DAta Mining & Exploration Program

β-release

Graphical User Interface User Manual

DAME-MAN-NA-0010

Issue: 1.0
Date: Dec 02, 2010
Author: M. Brescia

Doc. : BetaRelease_GUI_UserManual_DAME-MAN-NA-0010-Rel1.0

DAME Program
"we make science discovery happen"

INDEX

1 Introduction 5
2 Purpose 5
3 β-release GUI Overview 7
3.1 User Registration and Access 8
3.2 The command icons 10
3.3 Workspace Management 12
3.4 Header Area 13
3.5 Data Management 14
3.5.1 Upload user data 15
3.5.2 How to Create dataset files 17
3.5.2.1 Feature Selection 18
3.5.2.2 Column Ordering 20
3.5.2.3 Sort Rows by Column 22
3.5.2.4 Column Shuffle 23
3.5.2.5 Row Shuffle 25
3.5.2.6 Split by Rows 26
3.5.2.7 Dataset Scale 27
3.5.2.8 Single Column Scale 27
3.5.3 Download data 28
3.5.4 Moving data files 28
3.6 Experiment Management 29
3.6.1 Re-use of already trained networks 33

TABLE INDEX

Tab. 1 – Header Area Menu Options 14
Tab. 2 – Abbreviations and acronyms 38
Tab. 3 – Reference Documents 39
Tab. 4 – Applicable Documents 40

FIGURE INDEX

Fig. 1 – Suite functional hierarchy 7
Fig. 2 – The user registration/login form to access at the web application 8
Fig. 3 – The user registration form 9
Fig. 4 – An example of e-mail received by the user after submission of registration info 9
Fig. 5 – The Web Application starting main page (Resource Manager) 10
Fig. 6 – The Web Application main areas and commands 11
Fig. 7 – The right sequence to configure and execute an experiment workflow 12
Fig. 8 – the button "New Workspace" at left corner of workspace manager window 12
Fig. 9 – the form field that appears after pressing the "New Workspace" button 13
Fig. 10 – the active workspace created in the Workspace List Area 13
Fig. 11 – The GUI Header Area with all submenus open 14
Fig. 12 – The Upload data feature open in a new tab 15
Fig. 13 – The Upload data from external URI feature 16
Fig. 14 – The Upload data from Hard Disk feature 16
Fig. 15 – The Uploaded data file in the File Manager sub window 17
Fig. 16 – The dataset editor tab with the list of available operations 18
Fig. 17 – The Feature Selection operation – select columns and put saving name 19
Fig. 18 – The Feature Selection operation – the new file created 19
Fig. 19 – The Column Ordering operation – the starting view 20
Fig. 20 – The Column Ordering operation – new order to columns 21
Fig. 21 – The Column Ordering operation – new file created 21
Fig. 22 – The Sort Rows by Column operation – step 1 22
Fig. 23 – The Sort Rows by Column operation – step 2 23
Fig. 24 – The Sort Rows by Column operation – the new file created 23
Fig. 25 – The Column Shuffle operation – step 1 24
Fig. 26 – The Column Shuffle operation – the new file created 24
Fig. 27 – The Row Shuffle operation – step 1 25
Fig. 28 – The Row Shuffle operation – the new file created 25
Fig. 29 – The Split by Rows operation – step 1 26
Fig. 30 – The Split by Rows operation – the new files created 26
Fig. 31 – The Dataset Scale operation – step 1 27
Fig. 32 – The Dataset Scale operation – the new file created 27
Fig. 33 – The Single Column Scale operation – step 1 28
Fig. 34 – The Single Column Scale operation – the new file created 28
Fig. 35 – Creating a new experiment (by selecting icon "Experiment" in the workspace) 29
Fig. 36 – The new tab open after creation of a new experiment with the list of available options 30
Fig. 37 – The new state of the experiment configuration tab after the selection of the model 30
Fig. 38 – The configuration options in the Train use case 31
Fig. 39 – The configuration options in the Test use case 31
Fig. 40 – The configuration options in the Run use case 31
Fig. 41 – The configuration options in the Full use case 32
Fig. 42 – Example of a web page automatically open after the click on the help button 32
Fig. 43 – Some different state of two concurrent experiments 33
Fig. 44 – An example of Regression_MLP training case for the XOR problem 33
Fig. 45 – The status at the end of the XOR problem experiment 34
Fig. 46 – The list of output files after the XOR problem training experiment 34
Fig. 47 – The training error scatter plot downloaded from the experiment output list (x-axis is the training cycle, y-axis is the training mean square error) 35
Fig. 48 – The operation to "move" the trained network file in the Workspace input file list 36
Fig. 49 – the configuration for the Run use case in the XOR problem 36

Fig. 50 – the output of the Run use case experiment in the XOR problem *37*

1 Introduction

The present document is part of the DAMEWARE Web Application Suite (β release) user-side documentation package. The β release arises from the very primer version of the web application (α release) which has been made available to public domain since July 2010.
During last five months the developing team has spent much efforts to fix bugs, satisfy testing user requirements, suggestions and to improve the application features, by integrating several other data mining models, always coming from machine learning theory, which have been scientifically validated by applying them offline in several practical astrophysical cases (photometric redshifts, quasar candidate selection, globular cluster search, transient discovery etc). All cases dealing with time domain data rich astronomy. In this scenario, the α release has covered the role of an advanced prototype, useful to evaluate, tune and improve main features of the web application, basically in terms of:

- User friendliness: by taking care of the impact on new users, not necessarily expert in data mining or skilled in machine learning methodologies, by paying particular attention to the easiness of navigation through GUI options and to the learning speed in terms of experiment selection, preprocessing, setup and execution;
- Data I/O handling: easiness to upload/download data files, to edit and configure datasets from original data files and/or archives;
- Workspace handling: the capacity to create different work spaces, depending on the experiment type and data mining model choice;

Of course in the new release it was impossible to match all important and valid suggestions came from the α release testers. In principle not for bad will of developers, but mostly because in some cases, the requests would needed drastic re-engineering of some infrastructure components or simply because they went against our design requirements, issued at the very beginning of the project. Of course, this not implies necessarily that in next releases of the application these requests will not taken into account.
Anyway, we tried to satisfy as much as possible main requests concerning the improvement of ease to use. Also in terms of examples and guided tours in using the available models. Don't forget that neophyte users should spend a certain amount of time to read this and other manuals to learn their capabilities and usability topics before to move inside the application. This is particularly true in order to understand how to identify the right association of functionality domain and the data mining model to be applied to your own science case. But we recall that this is fully reachable by gaining experience with time and through several trial-and-error sessions.

2 Purpose

This manual is mainly dedicated to drive users through the GUI options and features. In other words to show how to navigate and to interact with the application interface in order to create working spaces, experiments, to upload/download and edit data files. We will stop our discussion here at level of configuration of the models, for which specific manuals are available. This in order to separate the use of the GUI from the theoretical implications related to the setup and use of available data mining models.
The access gateway to the β release, its complete documentation package and other resources is at the following address (page beta_info.html of the website):
http://voneural.na.infn.it/beta_info.html

Last pages of this document host tables with “Abbreviations & Acronyms”, “Reference” and “Applicable” document lists and the acknowledgments. All over the document the references are labeled as [Rxx] for “Reference” documents and [Axx] for “Applicable” documents (xx is the incremental index as reported in the list tables). “Applicable” documents are not public references (technical documents internal to the DAME working group) included for quick technical references. Users external to the working group may ask to consult (privately) these documents by e-mail, motivating the reasons. The complete list of the internal documentation is available at the following address of the program official website: http://voneural.na.infn.it/DAME_DOCUMENTATION_LIST.html.

Notes for readers

Further updates of this document are periodically produced. Please, check new versions publication on the β release access web page (http://voneural.na.infn.it/beta_info.html**).**

3 β-release GUI Overview

Main philosophy behind the interaction between user and the DMS (Data Mining Suite) is the following.
The DMS is a web application, accessible through a simple web browser. It is structurally organized under the form of working sessions (hereinafter named workspaces) that the user can create, modify and erase. You can imagine the entire DMS as a container of services, hierarchically structured as in Fig. 1. The user can create as many workspaces as desired and populate them with uploaded data files and with experiments (created and configured by using the Suite). Each workspace is enveloping a list of data files and experiments, the latter defined by the combination between a functionality domain and a series (one at least) of data mining models. From these considerations, it is obvious that a workspace makes sense if at least one data file is uploaded into. **So far, the first two actions, after logged in, are, respectively, to create a new workspace (by assigning it a name) and to populate it by uploading at least one data file, to be used as input for future experiments. The data file types allowed by the DMS are reported in the next sections.**
In principle there should be many experiments belonging to a single workspace, made by fixing the functional domain and by slightly different variants of a model setup and configuration or by varying the associated models.

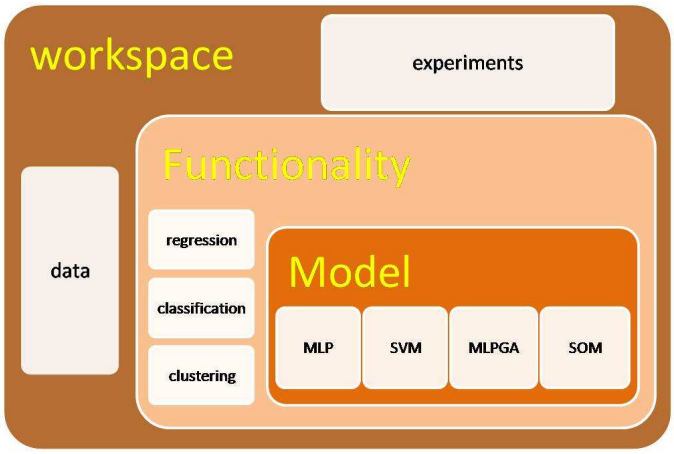


Fig. 1 – Suite functional hierarchy

By this way, as usual in data mining, the knowledge discovery process should basically consist of several experiments belonging to a specified functionality domain, in order to find the model, parameter configuration and dataset (parameter space) choices that give the best results (in terms of performance and reliability). The following sections describes in detail the practical use of the DMS from the end user point of view. Moreover, the DMS has been designed to build and execute a typical complete scientific pipeline (hereinafter named workflow) making use of machine learning models. This specification is crucial to understand the right way to build and configure data mining experiment with DMS.
In fact, machine learning algorithms (hereinafter named models) need always a pre-run stage, usually defined as training (or learning phase) and are basically divided into two categories: supervised and unsupervised models, depending, respectively, if they make use of a BoK (Base of Knowledge), i.e. couples input-target for each datum, to perform training or not (for more details about the concept of training data, see section 3.5 below).
So far, any scientific workflow must take into account the training phase inside its operation sequence.

Apart from the training step, a complete scientific workflow always includes a well-defined sequence of steps, including pre-processing (or equivalently preparation of data), training, validation, run, and in some cases post-processing.

The DMS permits to perform a complete workflow, having the following features:

- A workspace to envelope all input/output resources of the workflow;
- A dataset editor, provided with a series of pre-processing functionalities to edit and manipulate the raw data uploaded by the user in the active workspace (see section 3.5 for details);
- The possibility to copy output files of an experiment in the workspace to be arranged as input dataset for subsequent execution (the output of training phase should become the input for the validate/run phase of the same experiment);
- An experiment setup toolset, to select functionality domain and machine learning models to be configured and executed;
- Functions to visualize graphics and text results from experiment output;
- A plugin-based toolkit to extend DMS functionalities and models with user own applications;

3.1 User Registration and Access

The DMS makes use (embedded to the end user) of the Cloud computing infrastructure, made by single PCs in combination with GRID resources. This requires a reliable level of security in order to launch jobs (experiments) in a safe and coordinated way. This level of security is obtained by an accounting procedure that foresees an initial registration for new users, in order to activate their account on the DAME Suite. After activation, all subsequent accesses will require login and password, as defined by the user at the registration stage. The user registration/login entry page is shown in Fig. 2.

Fig. 2 – The user registration/login form to access at the web application

New users must be registered by following a very simple procedure requiring to select “Register Now” button on that page.
The registration form requires the following information to be filled in by the user (all fields are required):

- Name of the user;
- Family name of the user;
- User e-mail: the user e-mail (it will become his access login). It is important to define a real address, because it will be also used by the DMS for communications, feedbacks and activation instructions;
- Country: country of the user;
- Affiliation: the institute/academy/society of the user;
- Password: a safe password (at least 6 chars), without spaces and special chars;

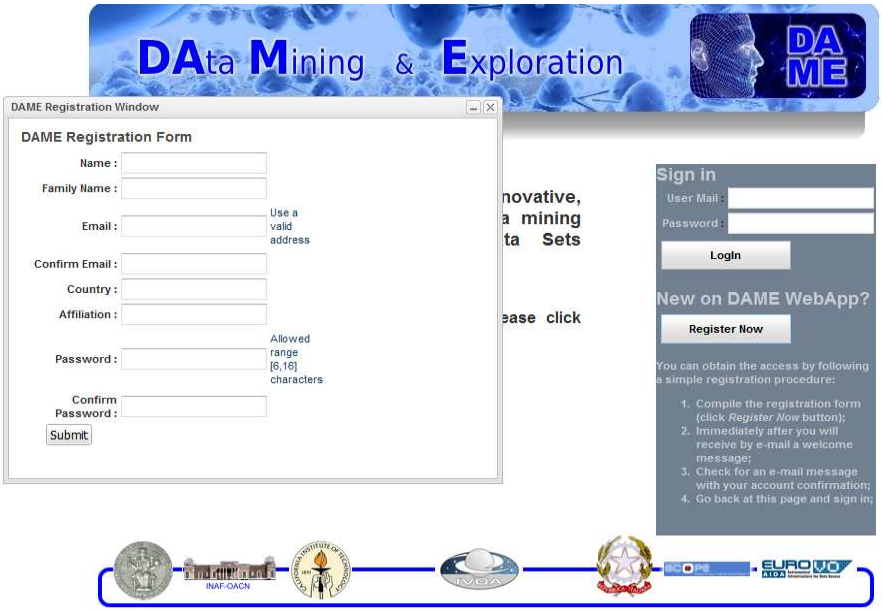


Fig. 3 – The user registration form

After submission, an e-mail will be immediately sent at the defined address (Fig. 4), confirming the correct coming up of the activation procedure.

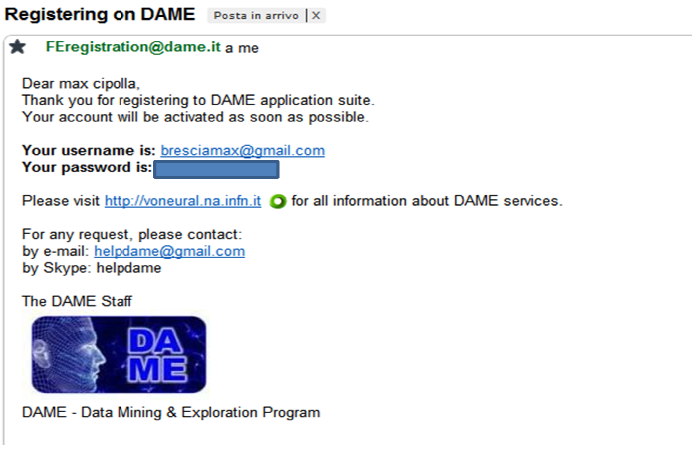


Fig. 4 – An example of e-mail received by the user after submission of registration info

After that the user must wait for a second e-mail which will be the final confirmation about the activation of the account. This is required in order to provide an higher security level.
Once the user has received the activation confirmation, he can access the webapp by inserting e-mail address and password.
The webapp will appear as shown in Fig. 5

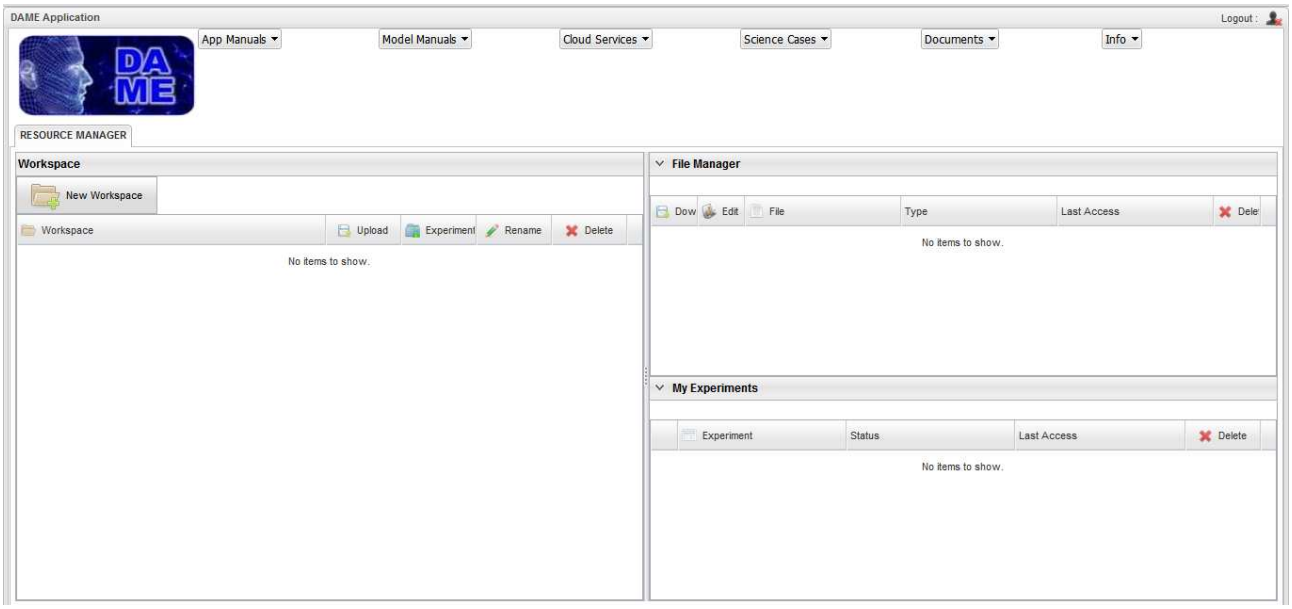


Fig. 5 – The Web Application starting main page (Resource Manager)

3.2 The command icons

The interaction between user and GUI is based on the selection of icons, which correspond to basic features available to perform actions. Here their description, related to the red circles in Fig. 6 is reported:

1. **The header menu options**. When one of the available menus is selected, a pop submenu appears with some options;
2. **Logout button**. If pressed the GUI (and related working session) is closed;
3. **Operation tabs**. The GUI is organized like a multi-tab browser. Different tabs are automatically open when user wants to edit data file to create/manipulate datasets, to upload files or to configure and launch experiments. All tabs can be closed by user, except the main one (Resource Manager);
4. **Creation of new workspaces**. When selected and named, the new workspace appears in the Workspace List Area (Workspace sub window);
5. **Workspace List Area**: portion of the main Resource Manager tab dedicated to host all user defined workspaces;
6. **Upload command**. When selected, the user is able to select a new file to be uploaded into the Workspace Data Area (Files Manager sub window). The file can be uploaded from external URI or from local (user) HD;
7. **Creation of new experiment**. When selected, the user is able to create a new experiment (a specific new tab is open to configure and launch the experiment);
8. **Rename workspace command**. When selected the user can rename the workspace;

9. **Delete Workspace command**. When selected, the user can delete the related workspace (only if no experiments are present inside, otherwise the system alerts to empty the workspace before to erase it);
10. **File Manager Area**: the portion of Resource Manager tab dedicated to list the data files belonging to various workspaces. All files present in this area are considered as input files for any kind of experiment;
11. **Download command**. When selected the user can download locally (on his HD) the selected file;
12. **Dataset Editor command**. When selected a new tab is open, where the user can create/edit dataset files by using all available dataset manipulation features;
13. **Delete file command**. When selected the user can delete the selected file from current workspace;
14. **Experiment List Area**: The portion of Resource Manager tab dedicated to the list of experiments and related output files present in the selected workspace;
15. **Experiment verbose list command**. When selected the user can open the experiment file list (for experiment in ended state) in a verbose mode, showing all related files created and stored;
16. **Delete Experiment command**: by clicking on it, the entire experiment (all listed files) is erased;
17. **Download experiment file command**. When selected the user can download locally (on his HD) the related experiment output file;
18. **AddinWS command**. When selected, the related file is automatically copied from the Experiment List Area to the currently active workspace File Manager Area. This feature is useful to re-use an output file of a previous experiment as input file of a new experiment (in the figure, look at the file weights.txt, that after this command is also listed in the File Manager). A file present in both areas, can be used as input either as output in the experiments.

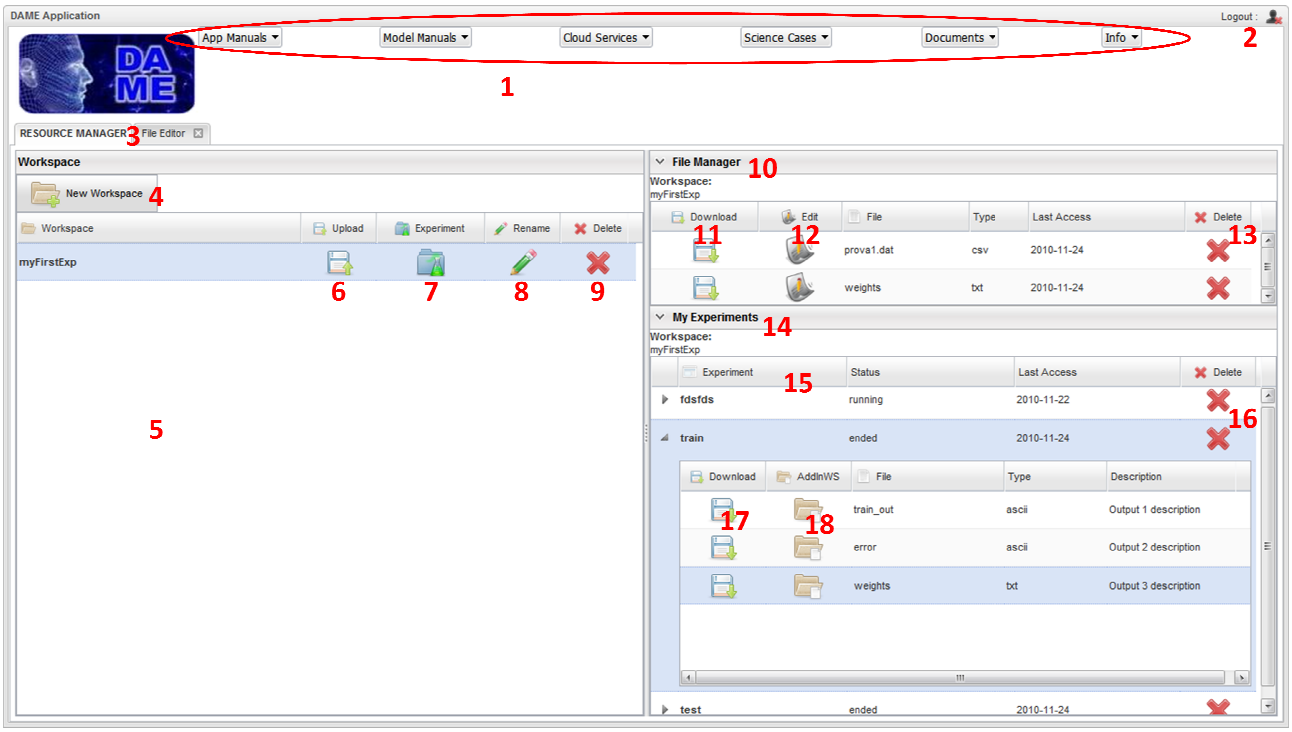


Fig. 6 – The Web Application main areas and commands

3.3 Workspace Management

A workspace is namely a working session, in which the user can enclose resources related to scientific data mining experiments. Resources can be data files, uploaded in the workspace by the user, files resulting from some manipulations of these data files, i.e. dataset files, containing subsets of data files, selected by the user as input files for his experiments, eventually normalized or re-organized in some way (see section 3.5 for details). Resources can also be output files, i.e. obtained as results of one or more experiments configured and executed in the current "active" workspace (see section 3.6 for details).
The user can create a new or select an existing workspace, by specifying its name. After opening the workspace, this automatically becomes the "active" workspace. This means that any further action, manipulating files, configuring and executing experiments, upload/download files, will result in the active workspace, Fig. 7. In this figure it is also shown the right sequence of main actions in order to operate an experiment (workflow) in the correct way.

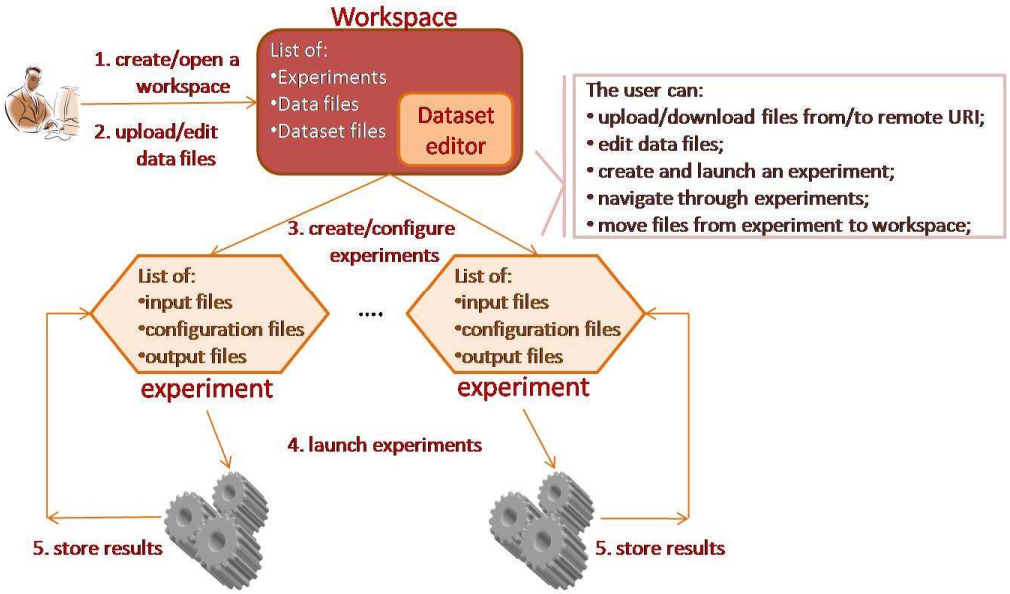


Fig. 7 – The right sequence to configure and execute an experiment workflow

So far, the basic role of a workspace is to make easier to the user the organization of experiments and related input/output files. For example the user could envelope in a same workspace all experiments related to a particular functionality domain, although using different models.
It is always possible to move (copy) files from experiment to workspace list, in order to re-use a same dataset file for multiple experiment sessions, i.e. to perform a workflow.
After access, the user must select the "active" workspace. If no workspaces are present, the user must create a new one, otherwise the user must select one of the listed workspace. The user can always create a new workspace by pressing the button as in Fig. 8.

Fig. 8 – the button "New Workspace" at left corner of workspace manager window

As consequence the user must assign a name to the new workspace, by filling in the form field as in Fig. 9.

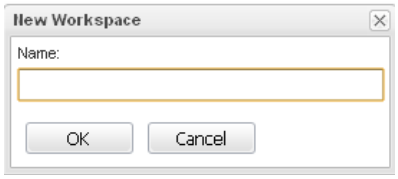


Fig. 9 – the form field that appears after pressing the "New Workspace" button

After creation, the active workspace can be populated by data and experiments, Fig. 10.

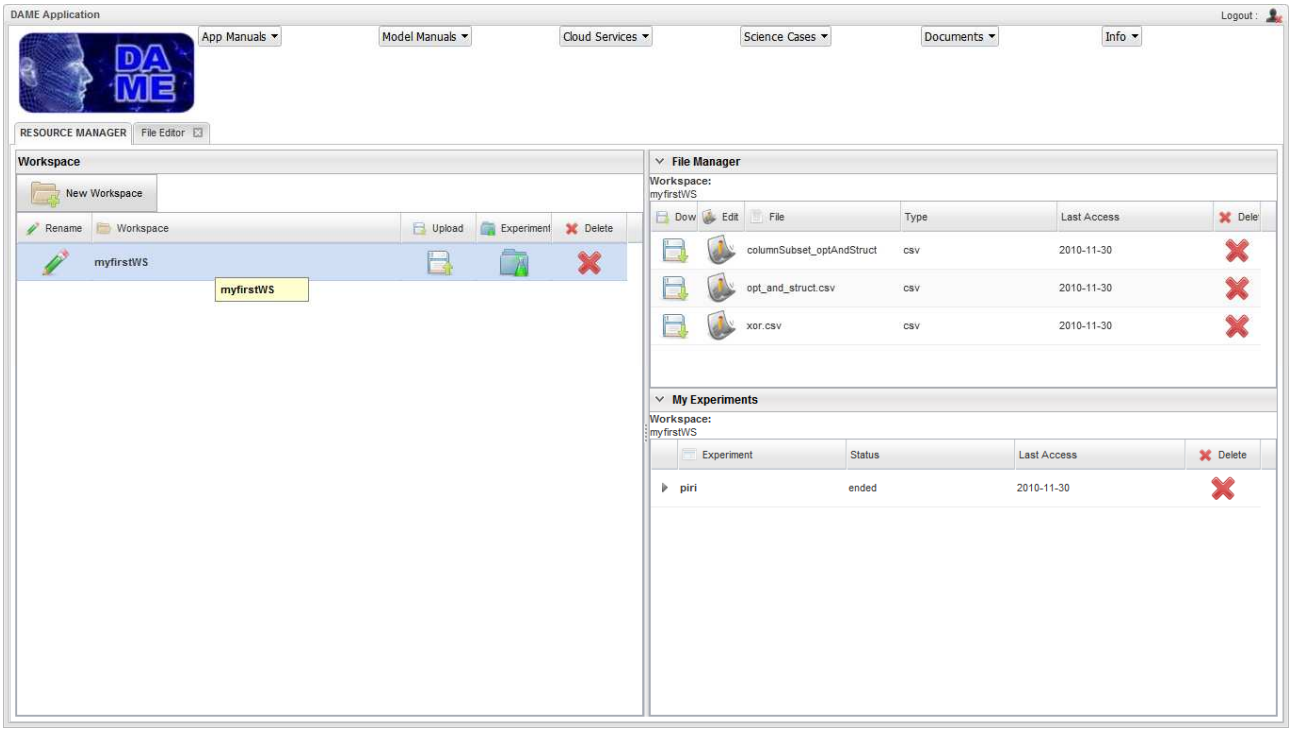


Fig. 10 – the active workspace created in the Workspace List Area

3.4 Header Area

At the top segment of the DMS GUI there is the so-called Header Area. Apart from the DAME logo, it includes a persistent menu of options directly related to information and documentation (this document also) available online and/or addressable through specific DAME program website pages.

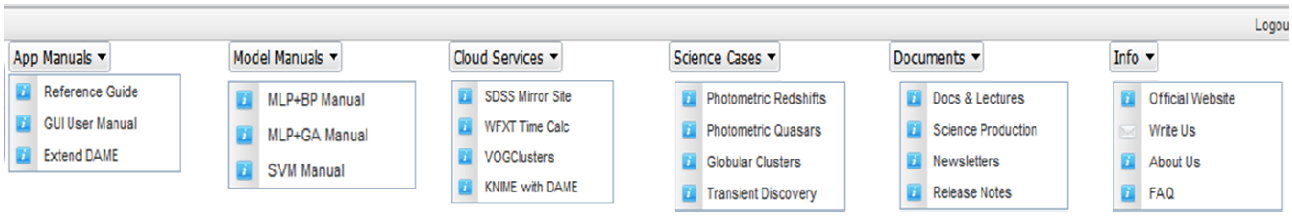


Fig. 11 – The GUI Header Area with all submenus open

The options are described in the following table (Tab. 1).

OPTIONS	HEADER	DESCRIPTION
Reference Guide	**Application Manuals**	http://voneural.na.infn.it/beta_info.html#manuals
GUI User Manual		
Extend DAME		http://voneural.na.infn.it/dmplugin.html
MLP+BP Manual	**Model Manuals**	Specific data mining model user manuals for experiments http://voneural.na.infn.it/beta_info.html#manuals
MLP+GA Manual		
SVM Manual		
SDSS Mirror Site	**Cloud Services**	http://dames.scope.unina.it/
WFXT Time Calc		http://voneural.na.infn.it/dame_wfxt.html
VOGCLUSTERS App		http://voneural.na.infn.it/vogclusters.html
KNIME with DAME		http://voneural.na.infn.it/dame_kappa.html
Photometric Redshifts	**Science Cases**	http://voneural.na.infn.it/vo_redshifts.html
Photometric Quasars		http://voneural.na.infn.it/qso.html
Globular Clusters		http://voneural.na.infn.it/dame_gcs.html
Transients Discovery		http://voneural.na.infn.it/dame_td.html
Docs & Lectures	**Documents**	http://voneural.na.infn.it/documents.html
Science Production		http://voneural.na.infn.it/science_papers.html
Newsletter		http://voneural.na.infn.it/newsletters.html
Release Notes		http://voneural.na.infn.it/beta_info.html#relnotes
Official website	**Info**	http://voneural.na.infn.it
Write Us		Mail to brescia@na.astro.it
About Us		http://voneural.na.infn.it/project_members.html
FAQ		http://voneural.na.infn.it/beta_info.html#faq

Tab. 1 – Header Area Menu Options

3.5 Data Management

The Data are the heart of the web application (data mining & exploration). All its features, directly or not, are involved within the data manipulation. So far, a special care has been devoted to features giving the opportunity to upload, download, edit, transform, submit, create data.
In the GUI input data (i.e. candidates to be inputs for scientific experiments) are basically belonging to a workspace (previously created by the user). All these data are listed in the "Files Manager" sub window. These data can be in one of the supported formats, i.e. data formats recognized by the web application as correct types that can be submitted to machine learning models to perform experiments. They are:

- **FITS (tabular .fits files);**
- **ASCII (.txt or .dat ordinary files);**

- **VOTable (VO compliant XML document files);**
- **CSV (Comma Separated Values .csv files);**

The user has to pay attention to use input data in one of these supported formats in order to launch experiments in a right way.

Other data types are permitted but not as input to experiments. For example, log, jpeg or "not supported" text files are generated as output of experiments, but only supported types can be eventually re-used as input data for experiments.
There is an exception to this rule for file format with extension **.ARFF (Attribute Relation File Format)**. These files can be uploaded and also edited by dataset editor, by using the type "CSV". But their extension .ARFF is considered "unsupported" by the system, so you can use any of the dataset editor options to change the extension (automatically assigned as CSV). Then you can use such files as input for experiments.
These output file are generally listed in the "Experiment Manager" sub window, that can be verbosely open by the user by selecting any experiment (when it is under "ended" state).
Other data files are created by dataset creation features, a list of operations that can be performed by the user, starting from an original data file uploaded in a workspace. These data files are automatically generated with a special name as output of any of the manipulation dataset operations available.
Confused? Well, don't panic please. Let's read carefully next sections.

3.5.1 Upload user data

As mentioned before, after the creation of at least one workspace, the user would like to populate the workspace with data to be submitted as input for experiments. Remember that in this section we are dealing with supported data formats only!

Fig. 12 – The Upload data feature open in a new tab

As shown in Fig. 12, when the user selects the "upload" command, (label nr. 6 in the Fig. 6), a new tab appears. The user can choose to upload his own data file from, respectively, from any remote URI (a priori known...!) or from his local Hard Disk.
In the first case (upload from URI), the Fig. 13 shows how to upload a supported type file from a remote address.

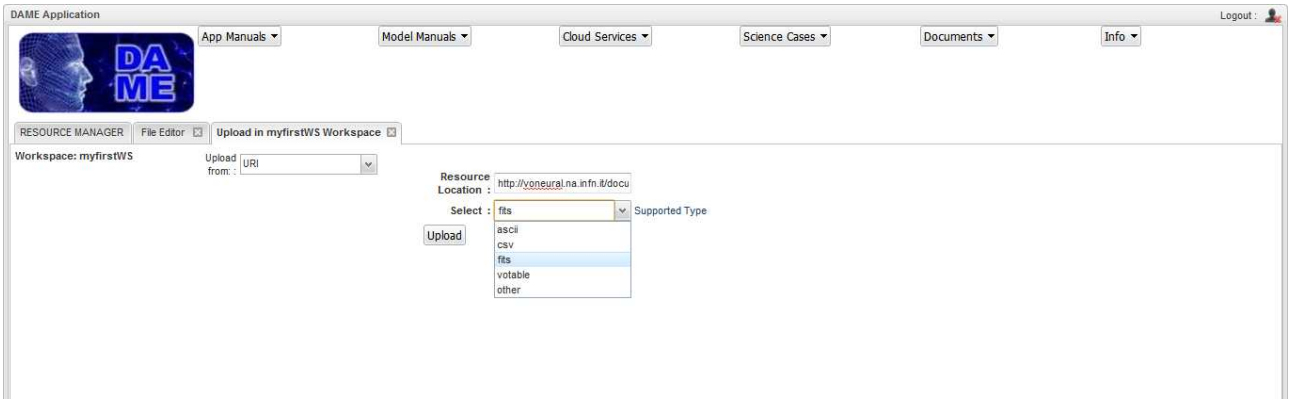


Fig. 13 – The Upload data from external URI feature

In the second case (upload from Hard Disk) the Fig. 14 shows how to select and upload any supported file in the GUI workspace from the user local HD.

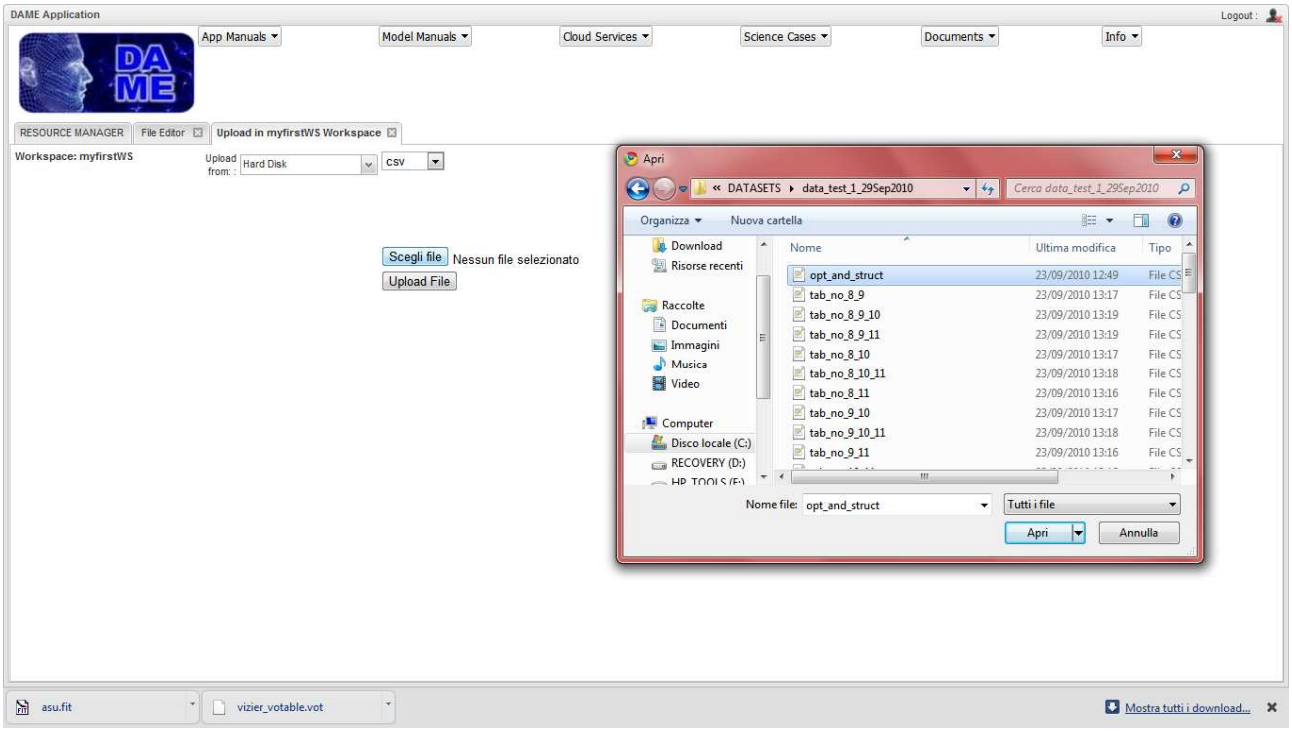


Fig. 14 – The Upload data from Hard Disk feature

After the execution of the operation, coming back to the main GUI tab, the user will found the uploaded file in the “Files Manager” sub window related with the currently active workspace, Fig. 15.

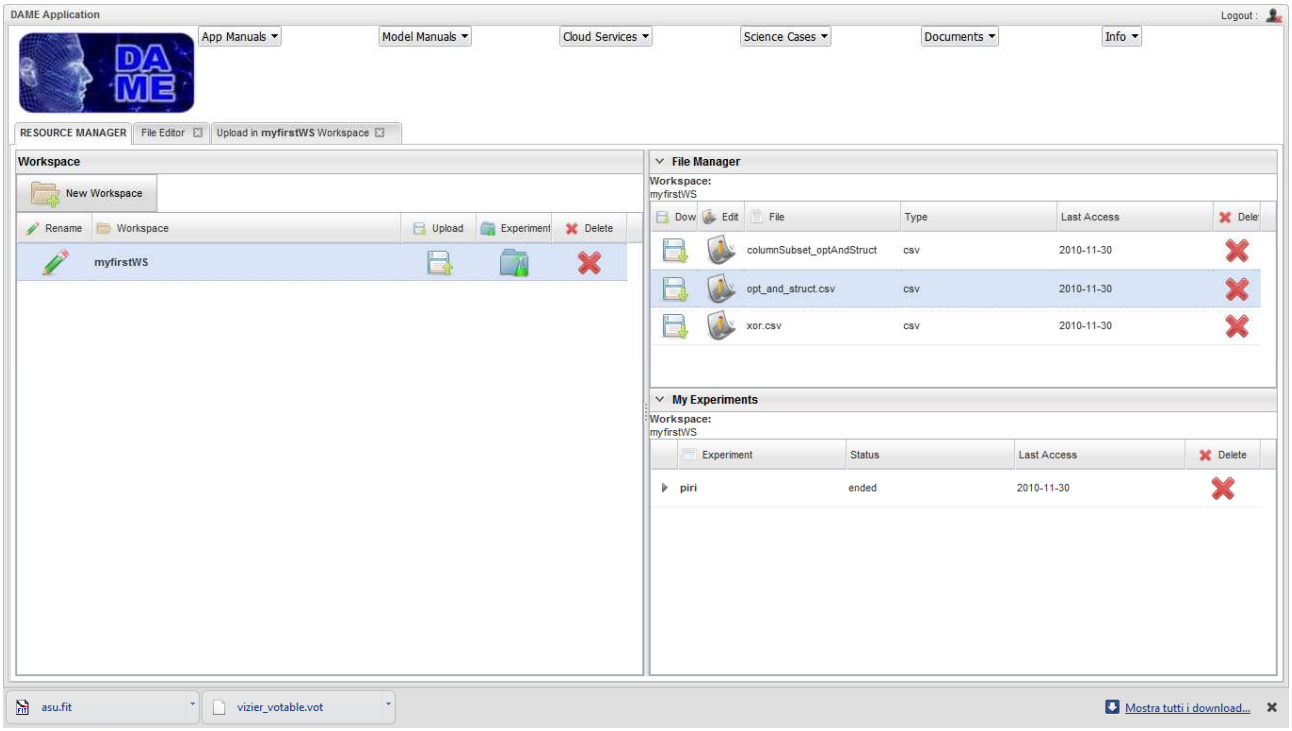


Fig. 15 – The Uploaded data file in the File Manager sub window

3.5.2 How to Create dataset files

If the user has already uploaded any supported data file in the workspace, it is possible to select it and to create datasets from it. This is a typical pre-processing phase in a machine learning based experiment, where, starting form an original data file, several different files must be prepared and provided to be submitted as input for, respectively, training, test and validate the algorithm chosen for the experiment. This pre-processing is generally made by applying one or more modification to the original data file (for example obtained from any astronomical observation run or cosmological simulation). The operations available in the web application are the following, Fig. 16:

- **Feature Selection;**
- **Columns Ordering;**
- **Sort Rows by Column;**
- **Column Shuffle;**
- **Row Shuffle;**
- **Split by Rows;**
- **Dataset Scale;**
- **Single Column Scale;**

All these operations, one by one, can be applied starting from a selected data file uploaded in the currently active workspace.

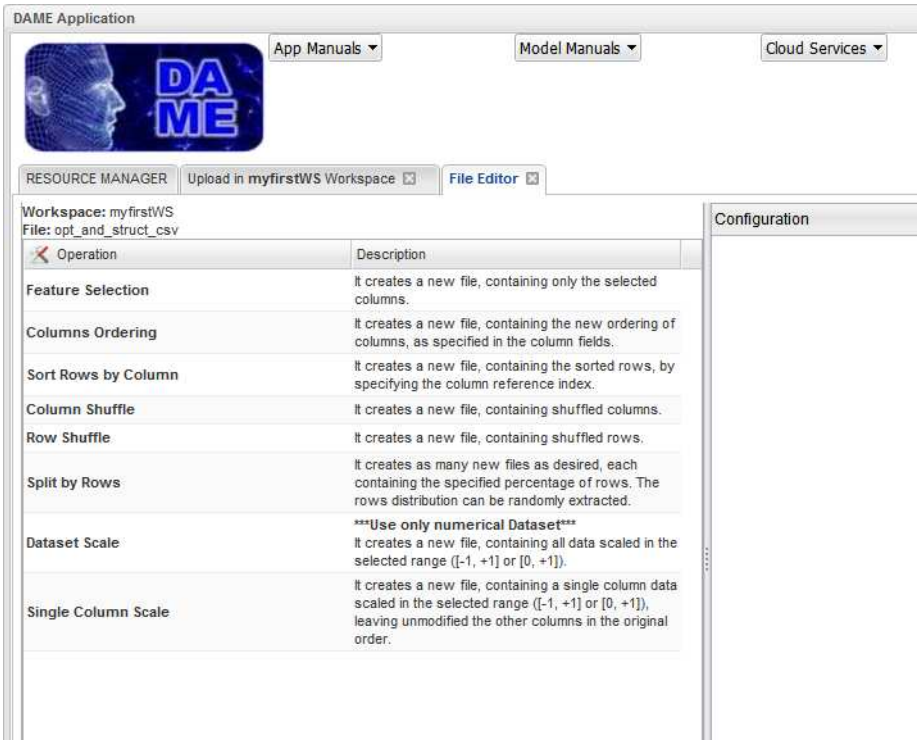


Fig. 16 – The dataset editor tab with the list of available operations

3.5.2.1 Feature Selection

This dataset operation permits to select and extract arbitrary number of columns, contained in the original data file, by saving them in a new file (of the same type and with the same extension of the original file), named as columnSubset_<user selected name> (i.e. with specific prefix *columnSubset*). This function is particularly useful to select training columns to be submitted to the algorithm, extracted from the whole data file. Details of the simple procedure are reported in Fig. 17 and Fig. 18.

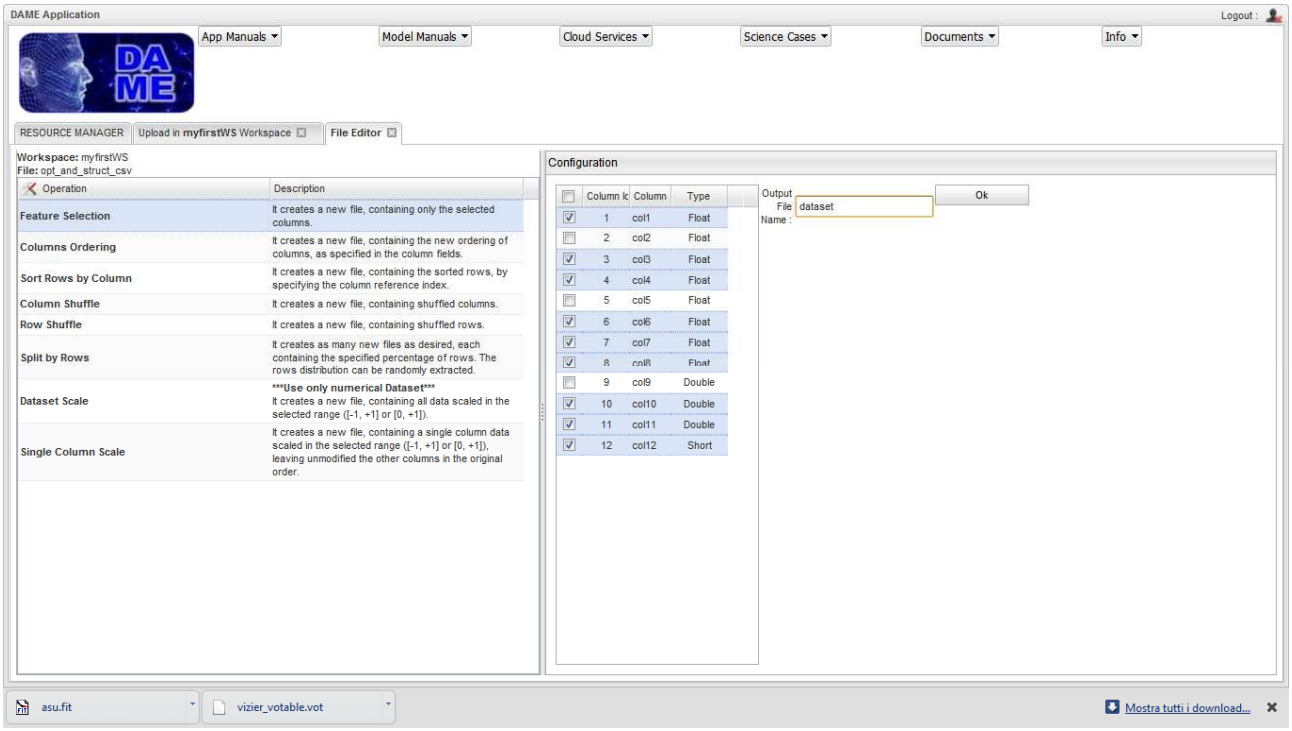


Fig. 17 – The Feature Selection operation – select columns and put saving name

As clearly visible in Fig. 17, the *Configuration* panel shows the list of columns originally present in the input data file, that can be selected by proper check boxes. Note that the whole content of the data file (in principle a massive data set) is not shown, but simply labelled by column meta-data (as originally present in the file).

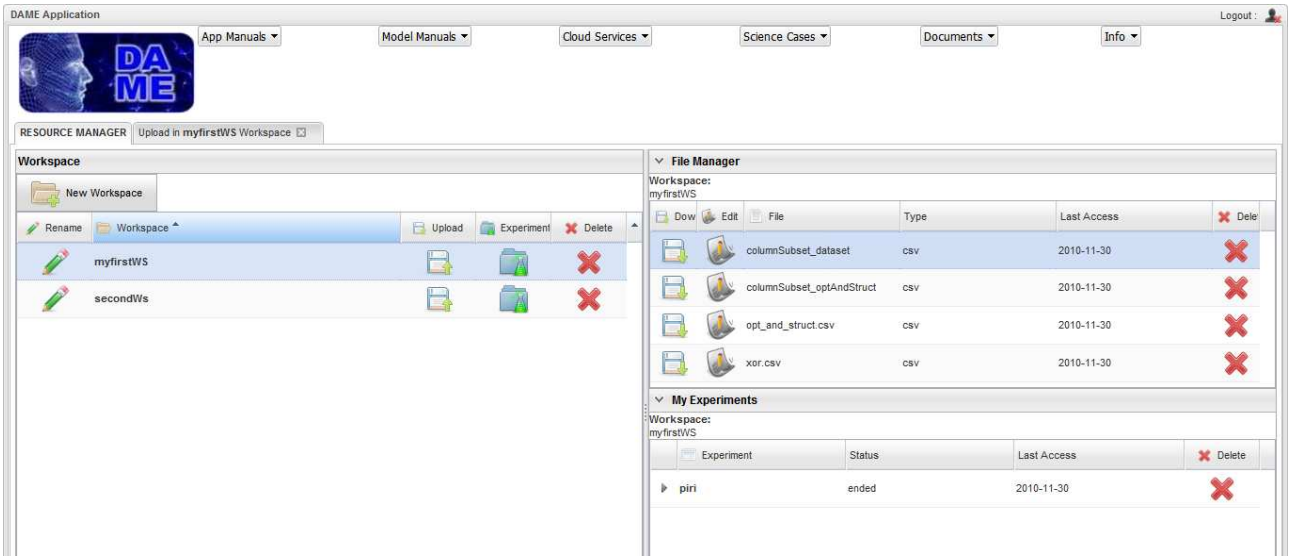


Fig. 18 – The Feature Selection operation – the new file created

3.5.2.2 Column Ordering

This dataset operation permits to select an arbitrary order of columns, contained in the original data file, by saving them in a new file (of the same type and with the same extension of the original file), named as columnSort_<user selected name> (i.e. with specific prefix *columnSort*). Details of the simple procedure are reported in Fig. 20 and **Errore. L'origine riferimento non è stata trovata.**.

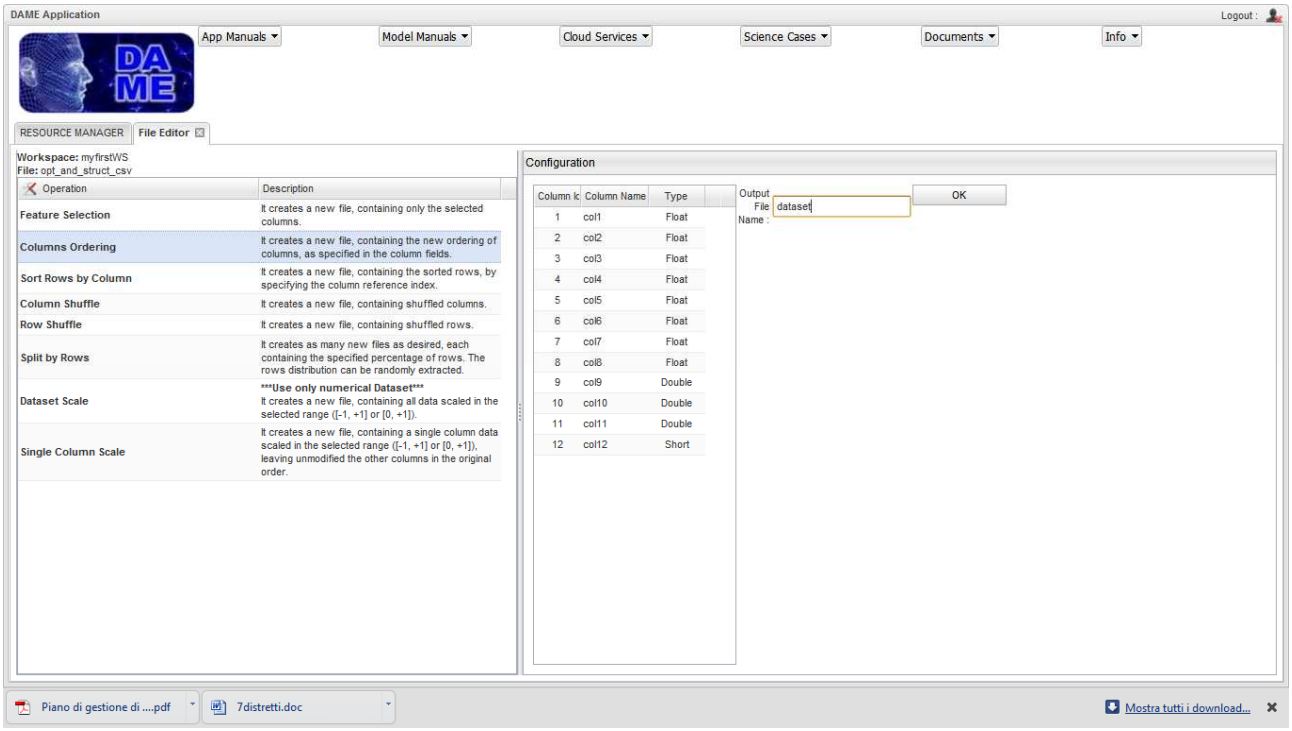


Fig. 19 – The Column Ordering operation – the starting view

In particular, in **Errore. L'origine riferimento non è stata trovata.** it is shown the result of several "dragging" operations operated on some columns. By selecting with mouse a column it is possible to drag it in a new desired position . At the end the new saved file will contain the new order given to data columns.

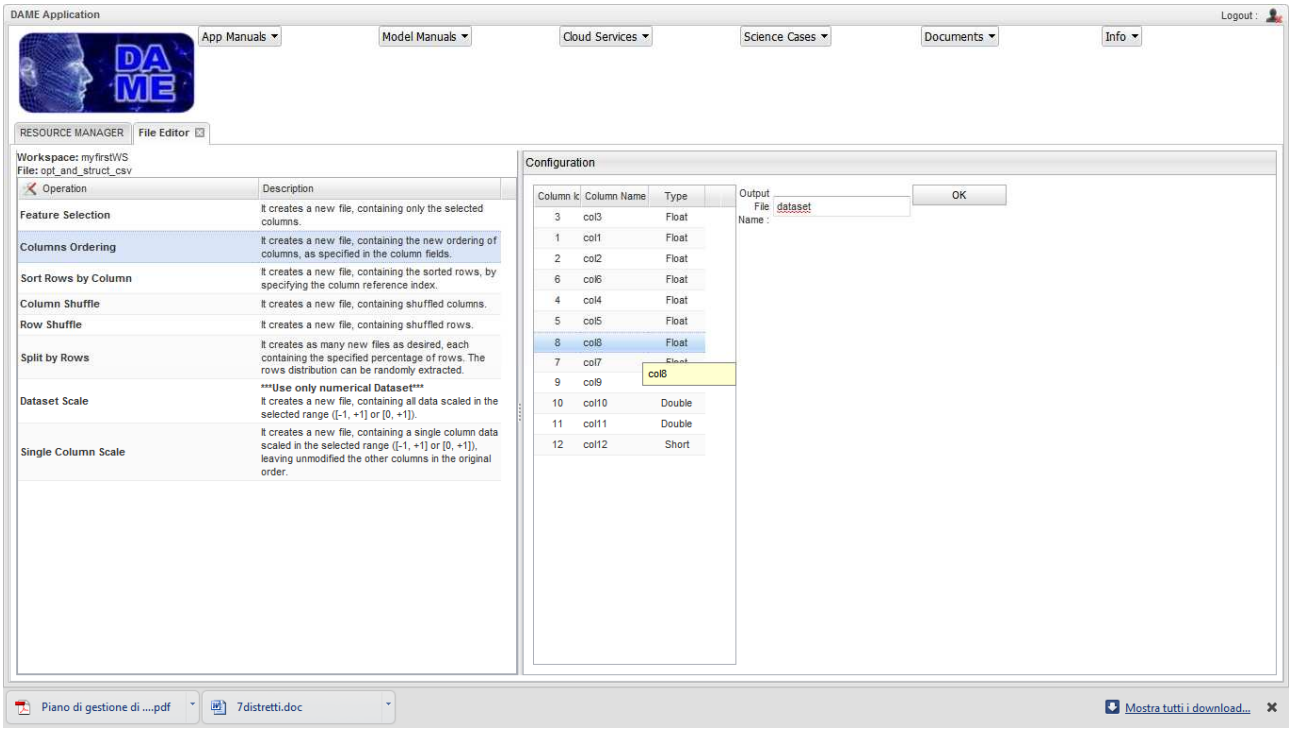


Fig. 20 – The Column Ordering operation – new order to columns

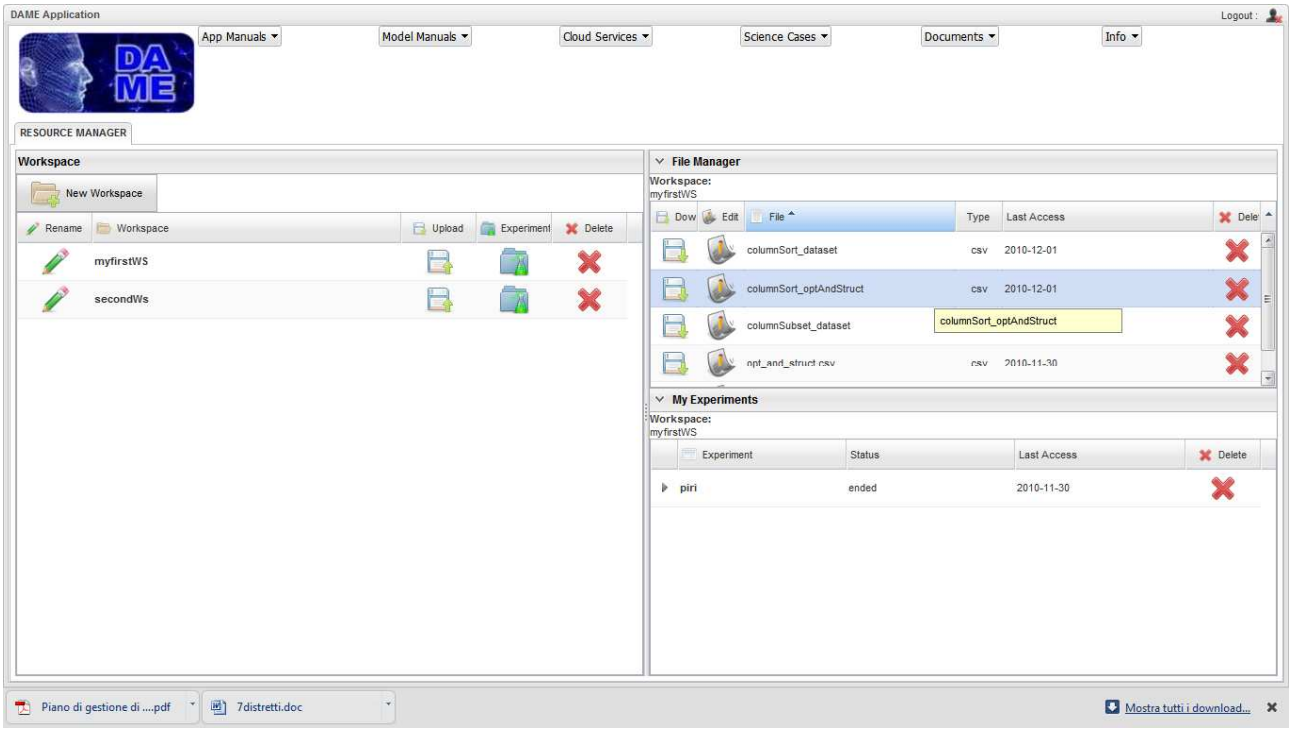


Fig. 21 – The Column Ordering operation – new file created

3.5.2.3 Sort Rows by Column

This dataset operation permits to select an arbitrary column, between those contained in the original data file, as sorting reference index for the ordering of all file rows. The result is the creation of a new file (of the same type and with the same extension of the original file), named as rowSort_<user selected name> (i.e. with specific prefix *rowSort*). Details of the simple procedure are reported in Fig. 22, Fig. 23 and Fig. 24.

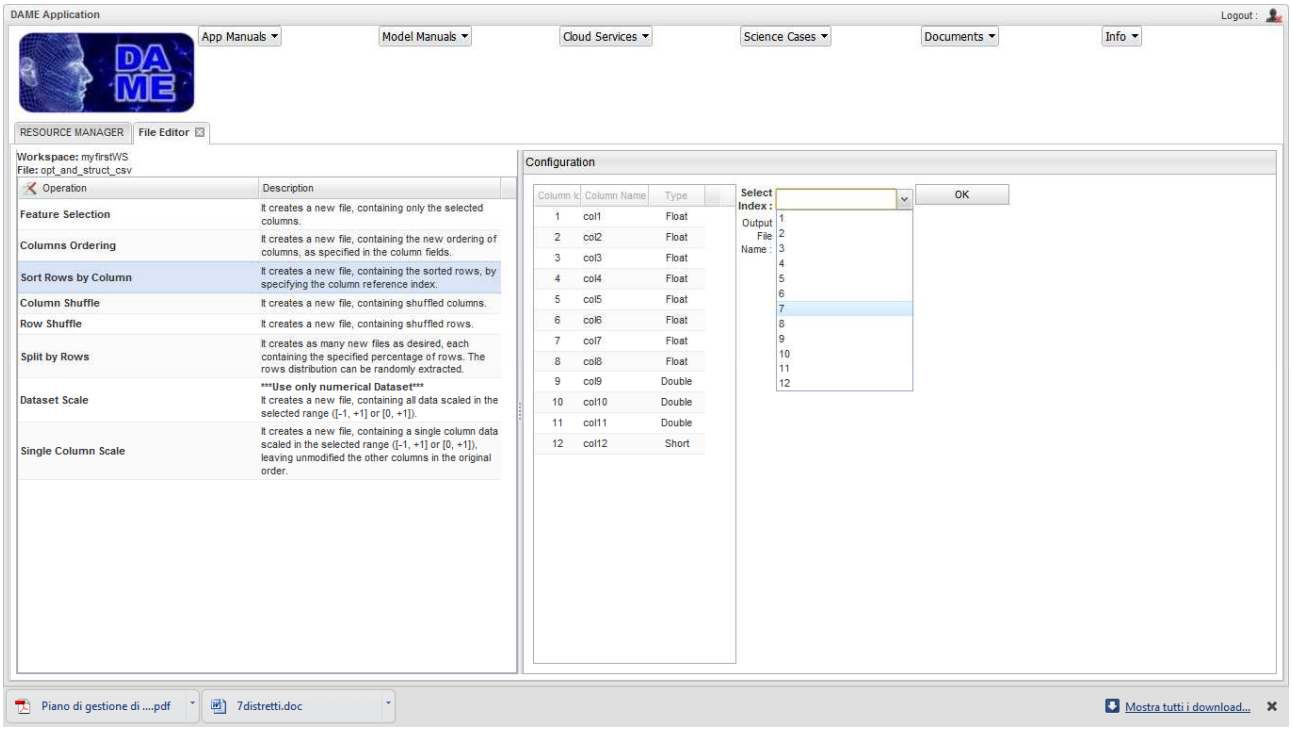


Fig. 22 – The Sort Rows by Column operation – step 1

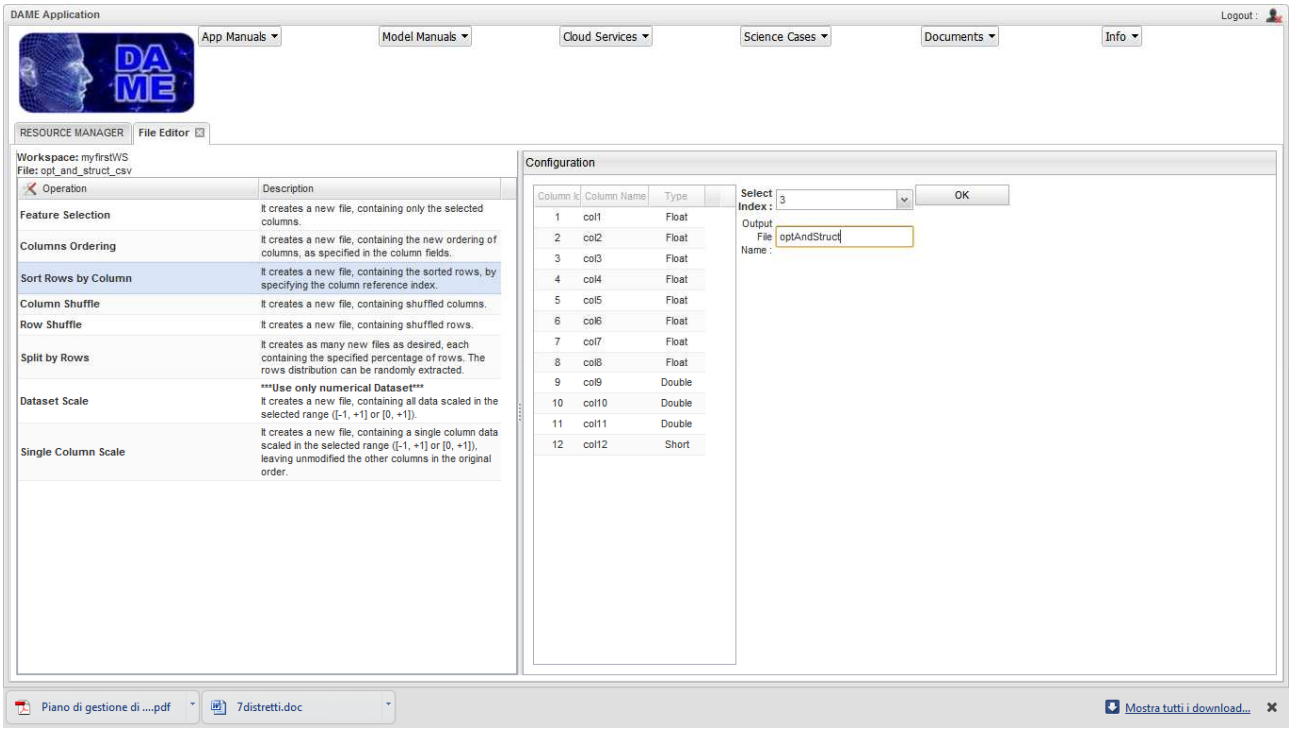


Fig. 23 – The Sort Rows by Column operation – step 2

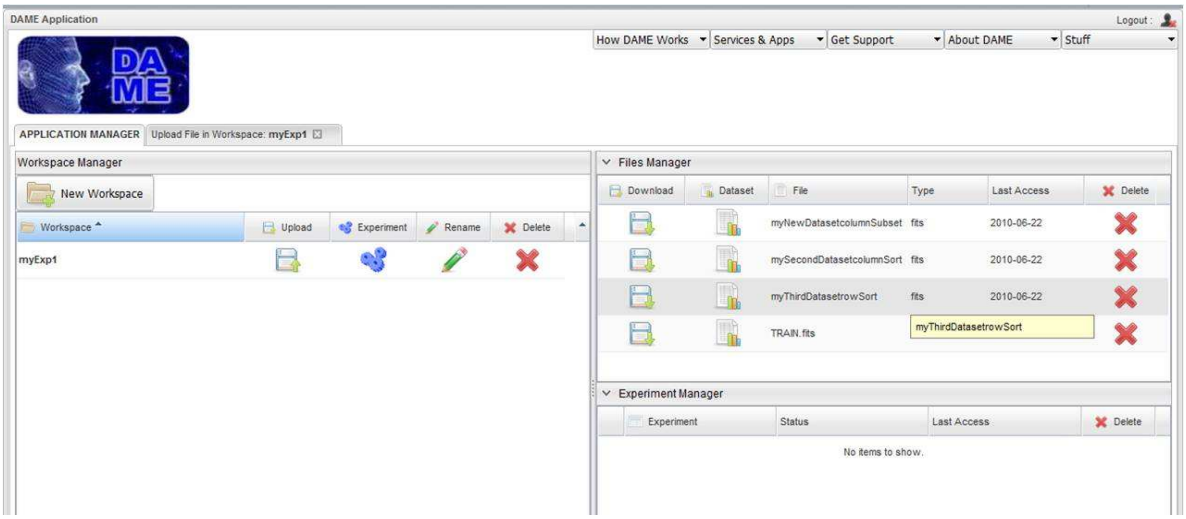


Fig. 24 – The Sort Rows by Column operation – the new file created

3.5.2.4 Column Shuffle

This dataset operation permits to operate a random shuffle of the columns, contained in the original data file. The result is the creation of a new file (of the same type and with the same extension of the original file), named as shuffle_<user selected name> (i.e. with specific prefix *shuffle*). Details of the simple procedure are reported in Fig. 25 and Fig. 26.

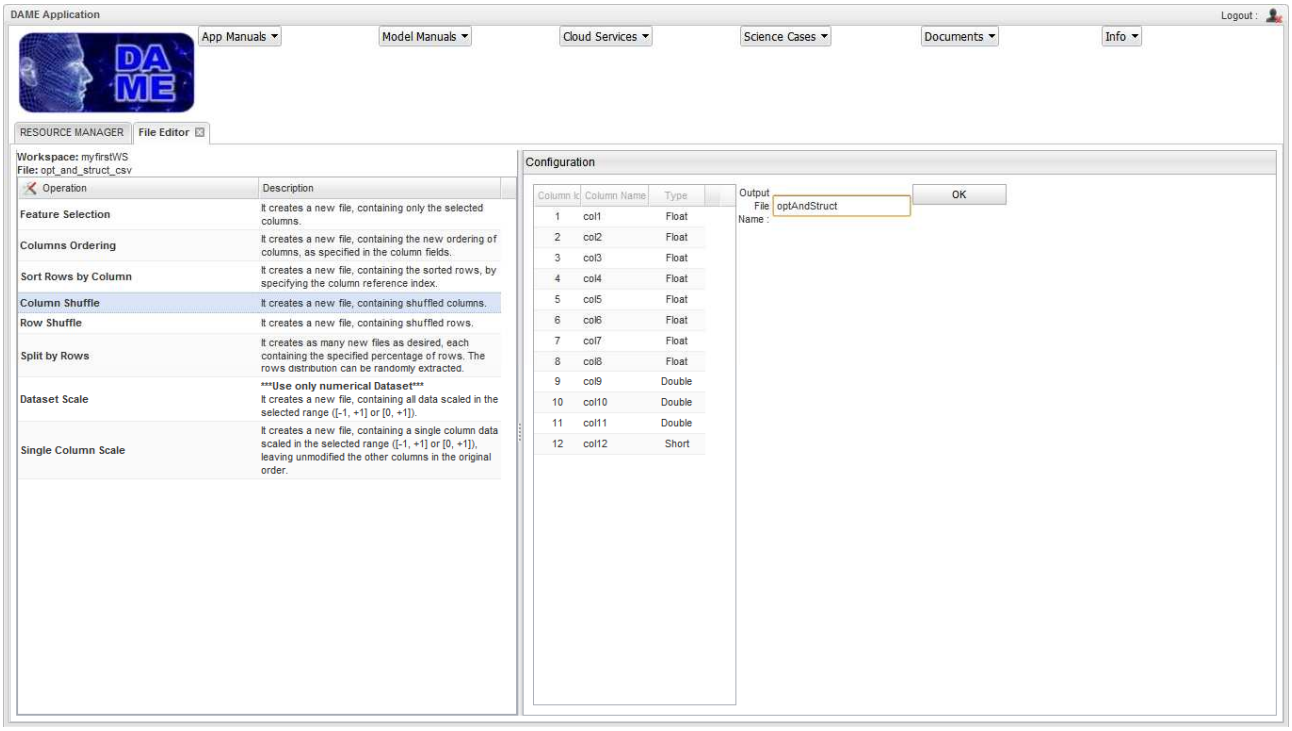


Fig. 25 – The Column Shuffle operation – step 1

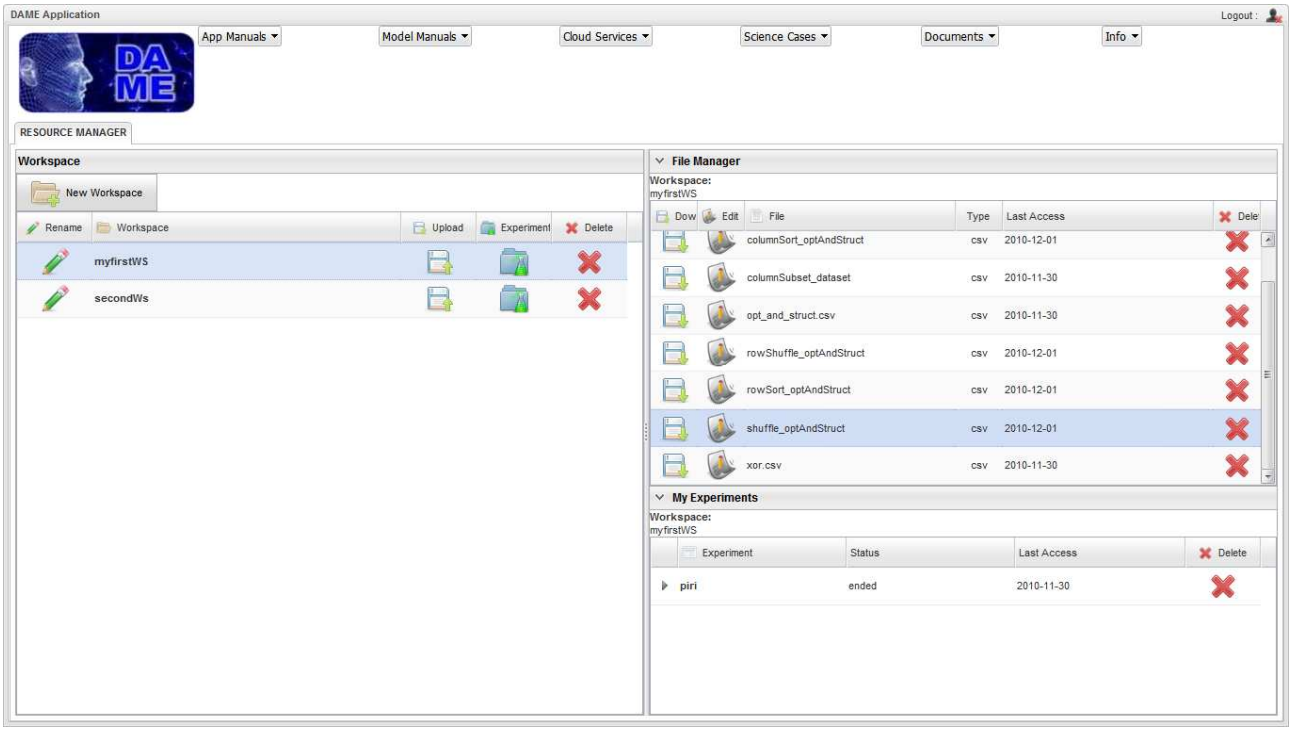


Fig. 26 – The Column Shuffle operation – the new file created

3.5.2.5 Row Shuffle

This dataset operation permits to operate a random shuffle of the rows, contained in the original data file. The result is the creation of a new file (of the same type and with the same extension of the original file), named as rowShuffle_<user selected name> (i.e. with specific prefix *rowShuffle*). Details of the simple procedure are reported in Fig. 27 and Fig. 28.

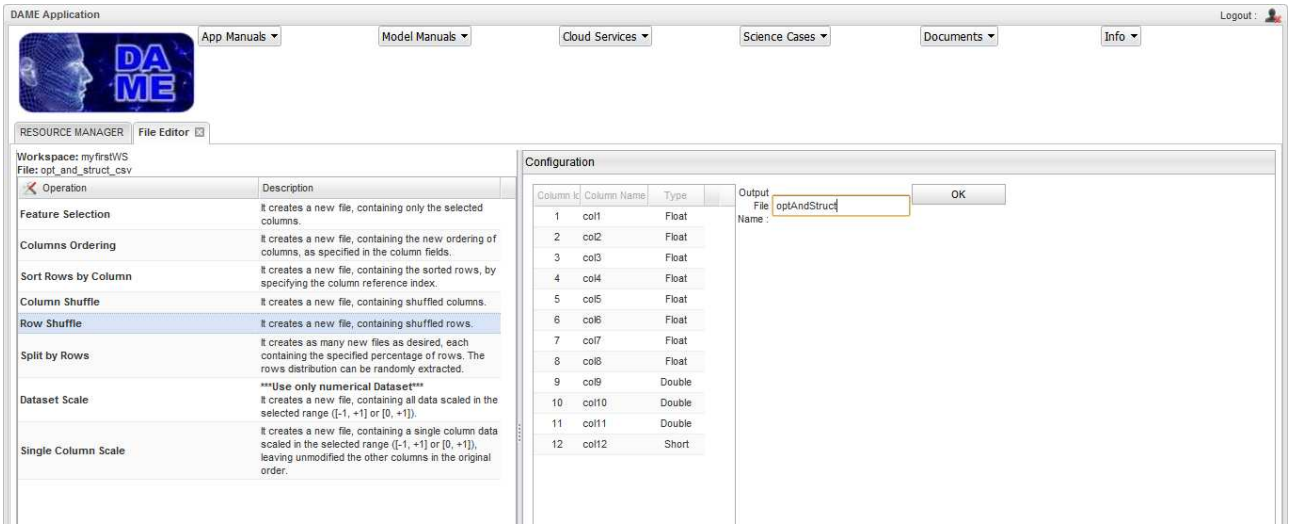


Fig. 27 – The Row Shuffle operation – step 1

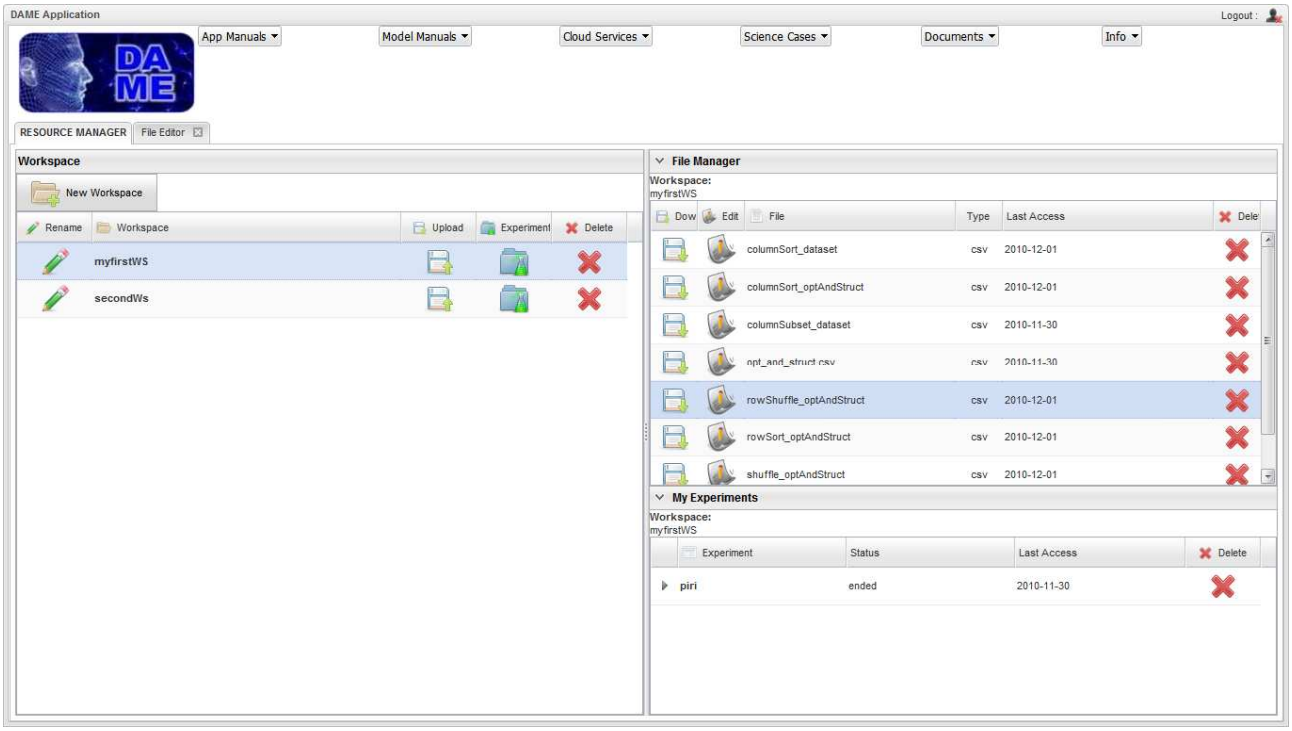


Fig. 28 – The Row Shuffle operation – the new file created

3.5.2.6 Split by Rows

This dataset operation permits to split the original file into two new files containing the selected percentages of rows, as indicated by the user. The user can move one of the two sliding bars in order to fix the desired percentage. The other sliding bar will automatically move in the right percentage position. The new file names are those filled in by the user in the proper name fields as split1_<user selected name> (split2_<user selected name>) (i.e. with specific prefix *split1*and *split2*). Details of the simple procedure are reported in Fig. 29, Fig. 30.

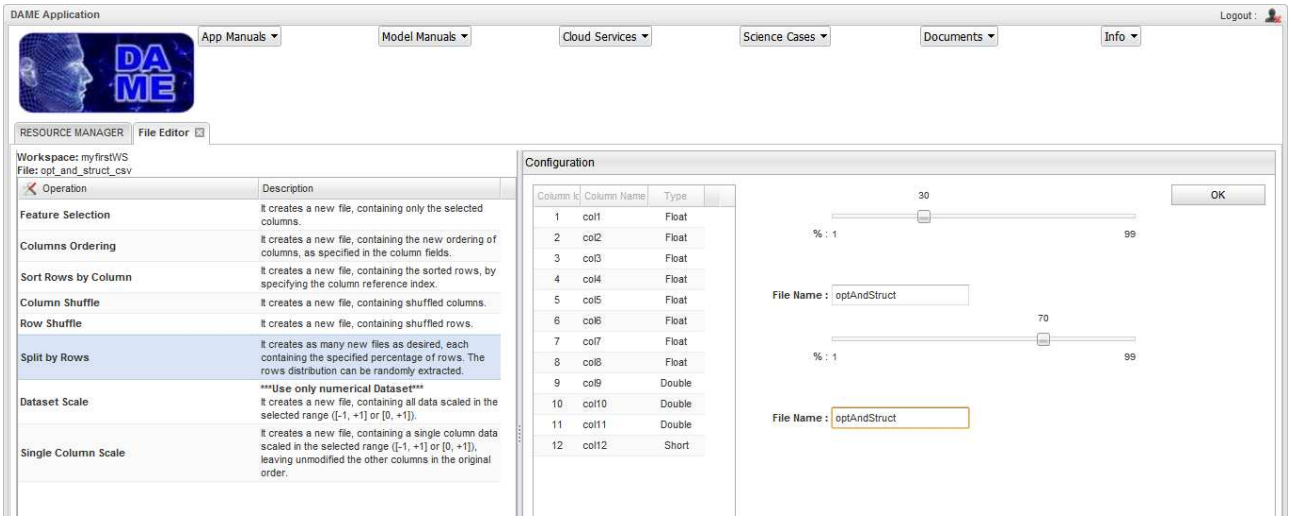


Fig. 29 – The Split by Rows operation – step 1

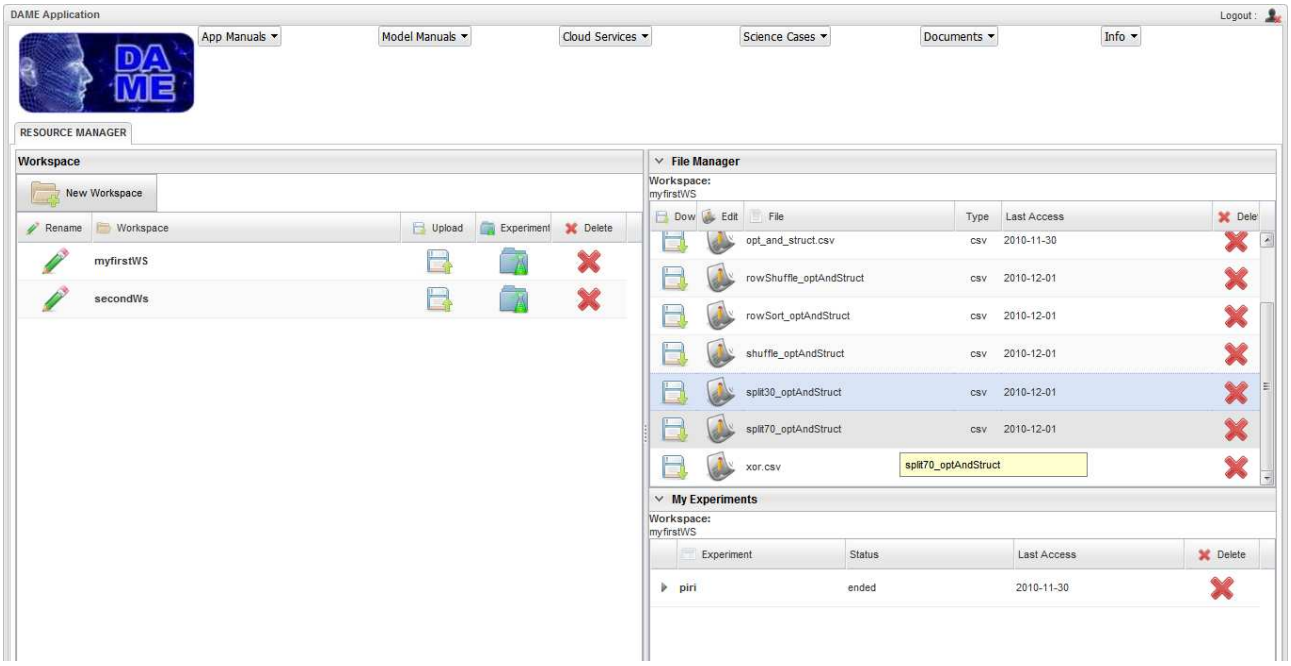


Fig. 30 – The Split by Rows operation – the new files created

3.5.2.7 Dataset Scale

This dataset operation (that works on numerical data files only!) permits to normalize column data in one of two possible ranges, respectively, [-1, +1] or [0, +1]. This is particularly frequent in machine learning experiments to submit normalized data, in order to achieve a correct training of internal patterns. The result is the creation of a new file (of the same type and with the same extension of the original file), named as scale_<user selected name> (i.e. with specific prefix *scale*). Details of the simple procedure are reported in Fig. 31 and Fig. 32.

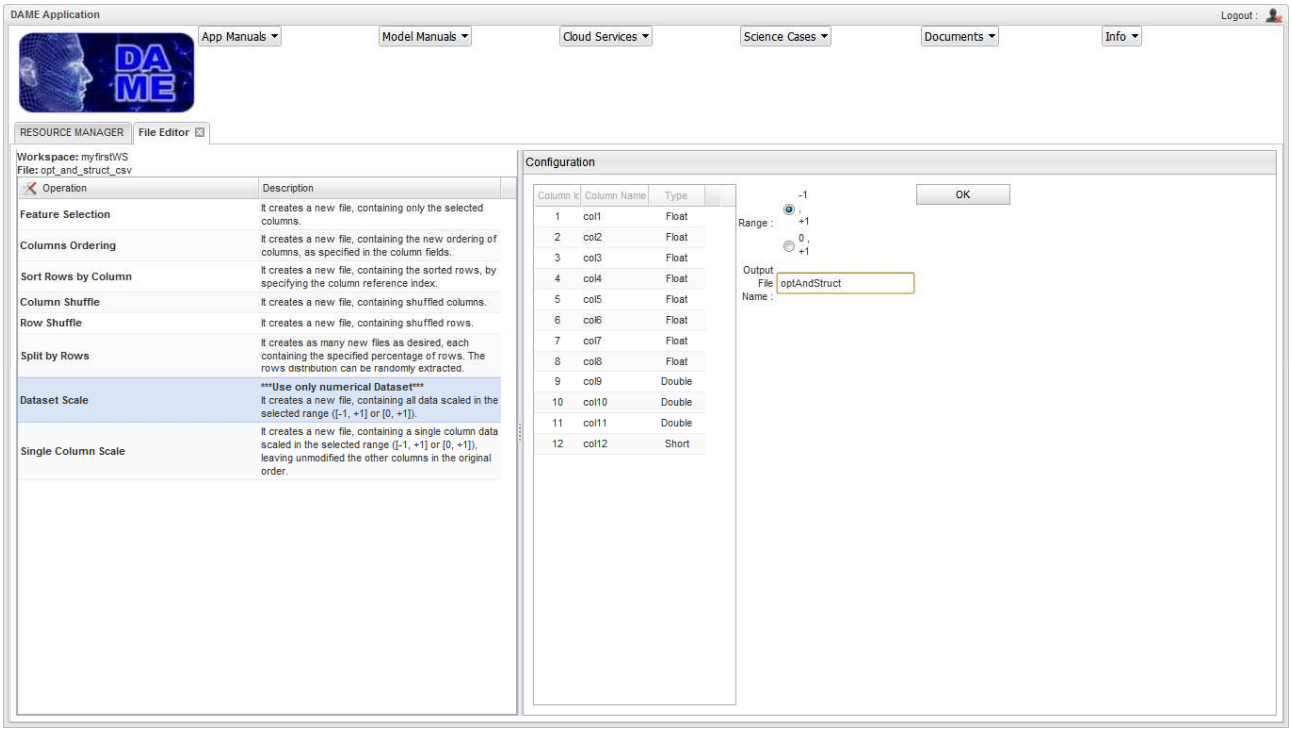


Fig. 31 – The Dataset Scale operation – step 1

Available in next releases

Fig. 32 – The Dataset Scale operation – the new file created

3.5.2.8 Single Column Scale

This dataset operation (that works on numerical data files only!) permits to normalize a single selected column, between those contained in the original file, in one of two possible ranges, respectively, [-1, +1] or [0, +1]. The result is the creation of a new file (of the same type and with the same extension of the original file), named as scaleOneCol_<user selected name> (i.e. with specific prefix *scaleOneCol*). Details of the simple procedure are reported in Fig. 33and Fig. 34.

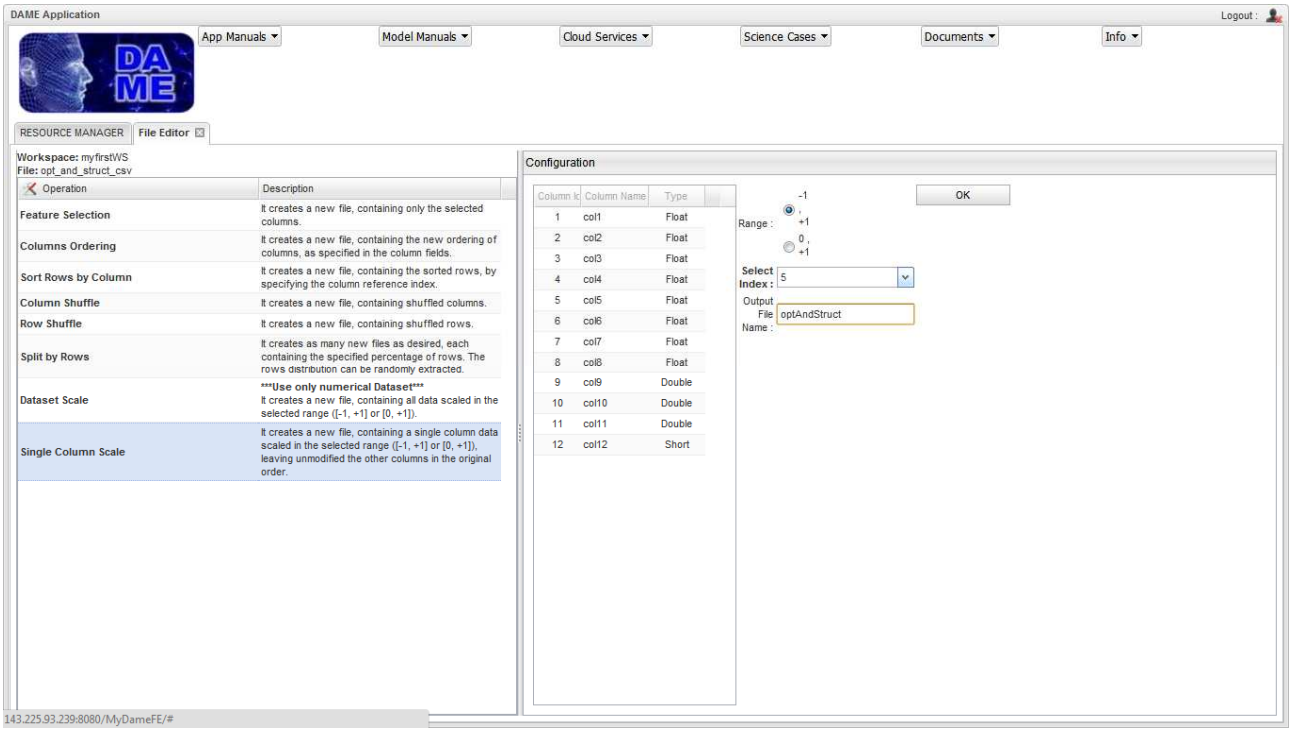


Fig. 33 – The Single Column Scale operation – step 1

Available in next releases

Fig. 34 – The Single Column Scale operation – the new file created

3.5.3 Download data

All data files (not only those of supported type) listed in the workspace and/or in the experiment panels, respectively, “Files Manager” and “Experiment Manager”, can be downloaded by the user on his own hard disk, by simply selecting the icon labelled with “Download” in the mentioned panels.

3.5.4 Moving data files

The virtual separation of user data files between workspace and experiment files, located in the respective panels (“File Manager” for workspace files, and “My Experiments” for experiment files), is due to the different origin of such files and depends on their registration policy into the web application database. The data files present in the workspace list (“File Manager” area panel) are usually registered as “input” files, i.e. to be submitted as inputs for experiments. While others, present in the experiment list (“My Experiments” panel), are considered as “output” files, i.e. generated by the web application after the execution of an experiment.
It is not rare, in machine learning complex workflows, to re-use some output files, obtained after training phase, as inputs of a test/validation phase of the same workflow. This is true for example for a MLP weight matrix file, output of the training phase, to be re-used as input weight matrix of a test (or validation) session of the same network.

In order to make available this fundamental feature in our application, the icon command nr. 18 (AddInWS) in Fig. 6, associated to each output file of an experiment, can be selected by the user in order to “copy” the file from experiment output list to the workspace input list, becoming immediately available as input file for new experiments belonging to the **same** workspace**: as important remark, in the beta release it is not yet possible to “move” files from a workspace to another**. The alternative procedure to perform this action is to download the file on user local Hard Disk and to upload it into another desired workspace in the webapp.

3.6 Experiment Management

After creating at least one workspace, populating it with input data files (of supported type) and optionally creating any dataset file, the next logical operation required is the configuration and launch of an experiment. In what follows, we will explain the experiment configuration and execution by making use of an example (very simple not linearly separable XOR problem) which can be replicated by the user by using the xor.csv and xor_run.csv data files (downloadable from the beta intro web page, http://voneural.na.infn.it/beta_info.html).

The Fig. 35 shows the initial step required, i.e. the selection of the icon command nr. 7of Fig. 6 in order to create the new experiment.

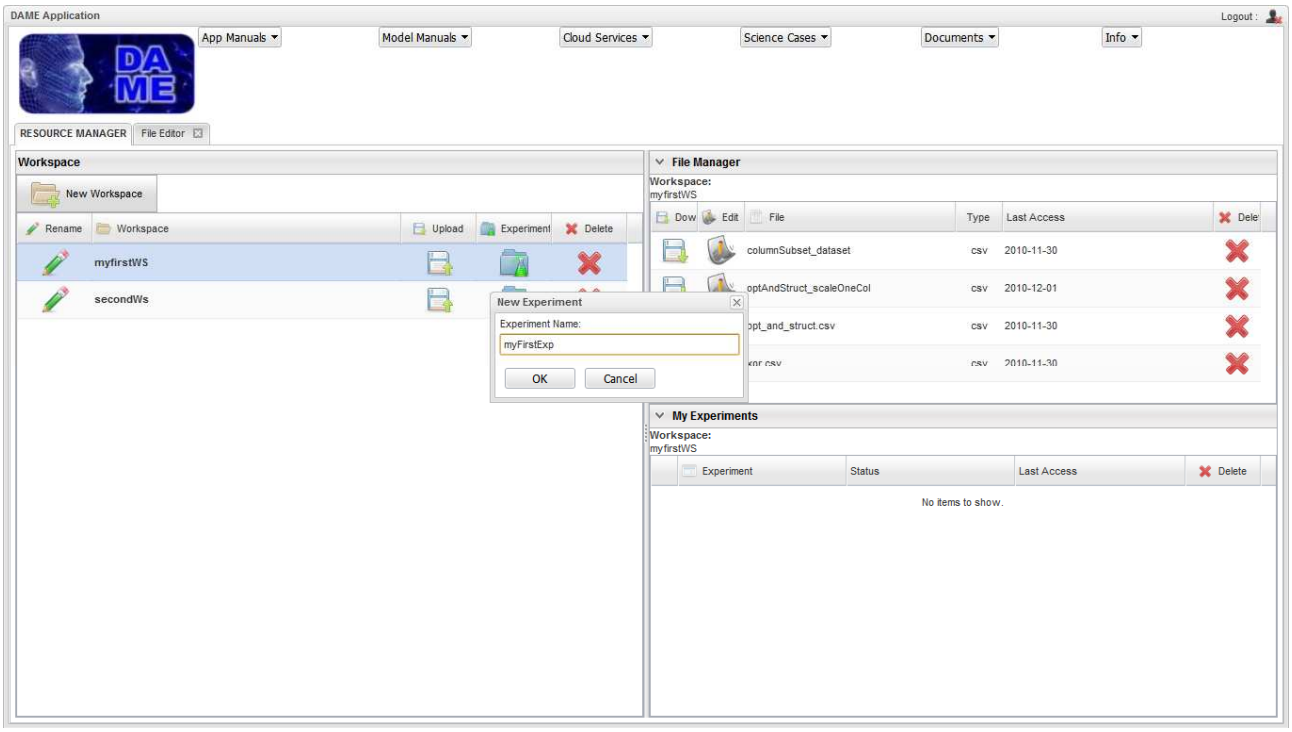


Fig. 35 – Creating a new experiment (by selecting icon “Experiment” in the workspace)

Immediately after, an automatic new tab appears, making available all basic features to select, configure and launch the experiment. In particular there is the list of couples [functionality]-[model] to choose for the current experiment. The proper choice should be done in order to solve a particular problem. It depends basically on the dataset to be used as input and on the output the user wants to obtain. Please, refer to the particular model reference manual for more details.

Fig. 36 – The new tab open after creation of a new experiment with the list of available options

The user can choose between classification or regression type of functionality to be applied to his problem. Each of these functionalities can be achieved by associating a particular data mining model, chosen between three types:

- MLP : Multilayer Perceptron neural network trained by standard Back Propagation (descent gradient of the error) learning rule;
- MLPGA: Multilayer Perceptron neural network trained by Genetic Algorithm learning rule;
- SVM: Support Vector Machine model;

Specific related manuals are available to obtain detailed information about the use of the above models (see webapp header menu options).

After the selection of the proper functionality-model, the tab will show (greyed) some options and the possibility to select the use case. The greyed options (like help button) will be activated after the selection of the use case to be configured and launched.

Fig. 37 – The new state of the experiment configuration tab after the selection of the model

As known, data mining models, following machine learning paradigm, offer a series of use cases (see figures below):

- **Train**: training (learning) phase in which the model is trained with the user available BoK;

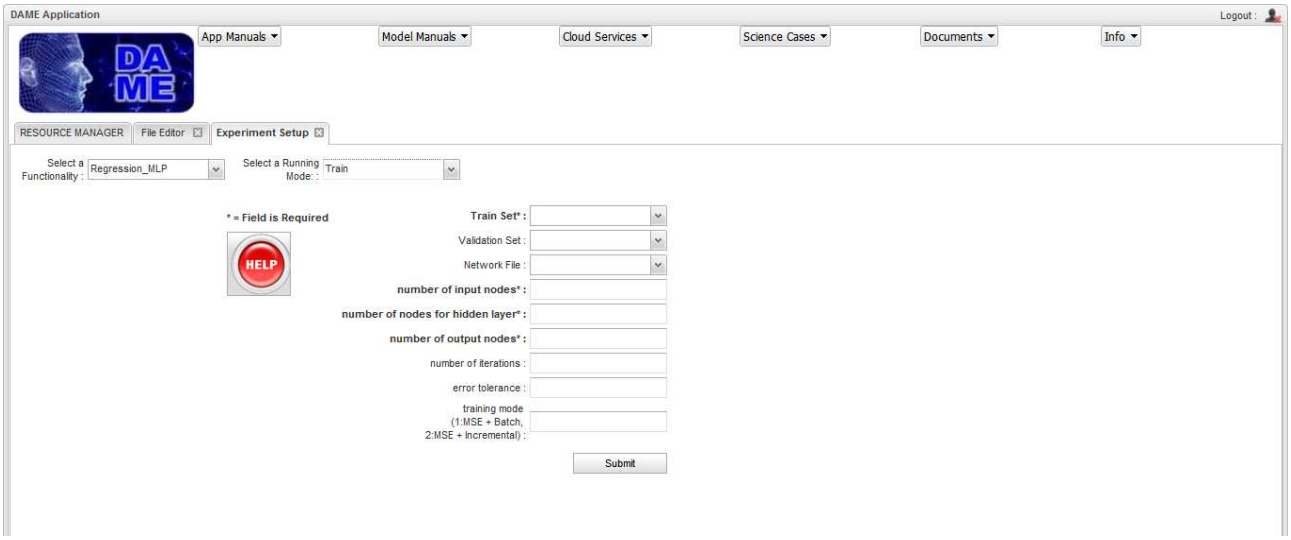


Fig. 38 – The configuration options in the Train use case

- **Test**: a sort of validation of the training phase. It can done by submitting the same training dataset, or a subset or a mix between already submitted and new dataset patterns;

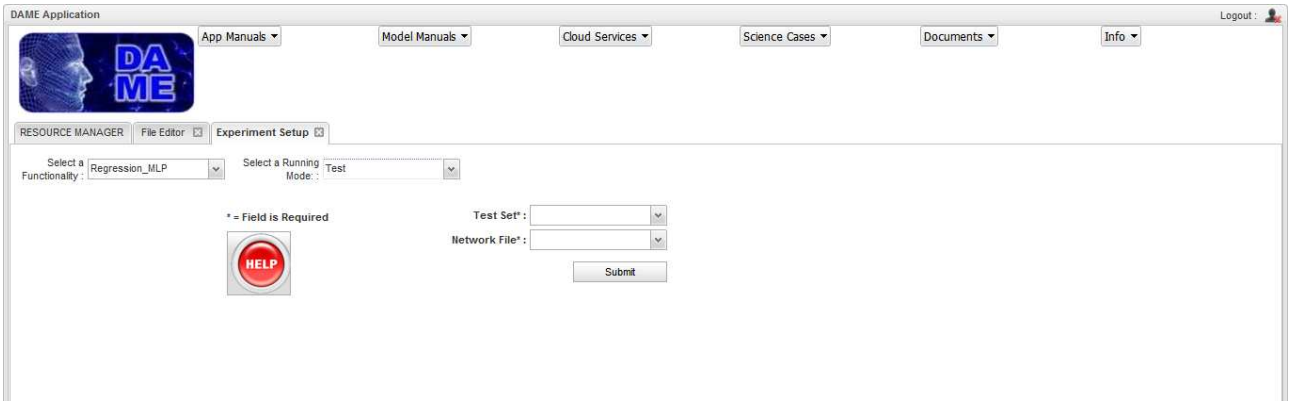


Fig. 39 – The configuration options in the Test use case

- **Run**: normal use of the already trained model;

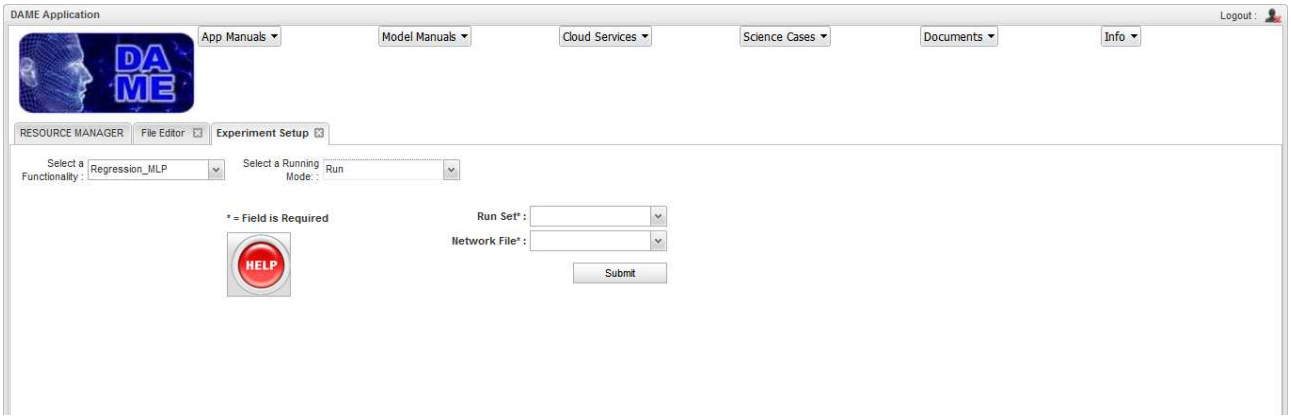


Fig. 40 – The configuration options in the Run use case

- **Full**: the complete and automatic serialized execution of the three previous use cases (train, test and Run). It is a sort of workflow, considered as a complete and exhaustive experiment for a specific problem.

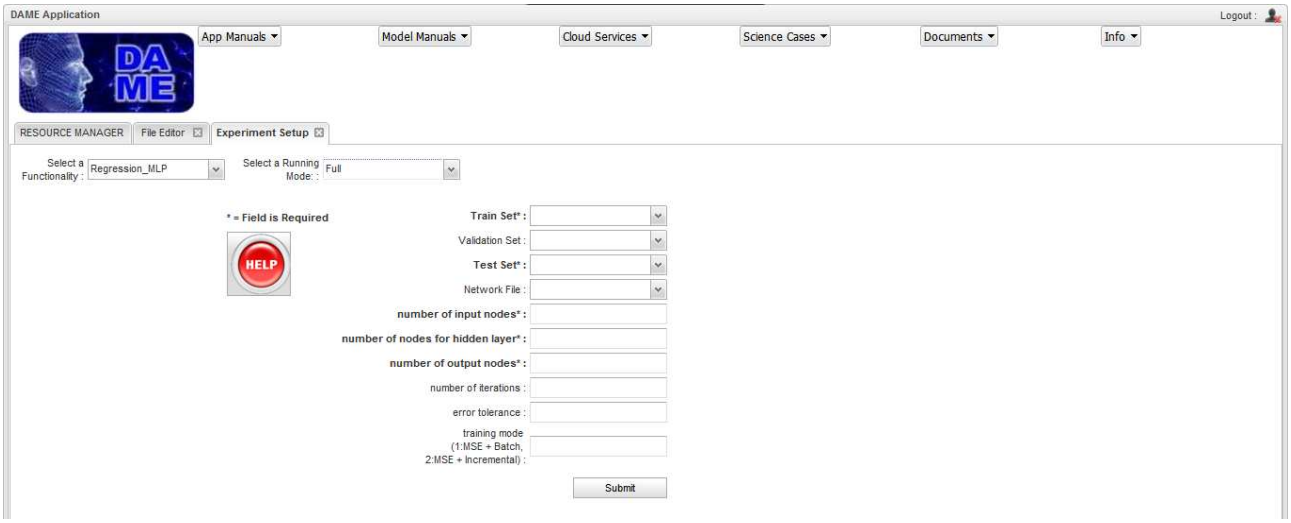


Fig. 41 – The configuration options in the Full use case

In all the above use case tabs, the help button redirects to a specific web page, reporting in verbose mode detailed description of all parameters. In particular, the parameter fields marked by an asterisk are considered "required" by the user. All other parameters can be left empty, by assuming a default value (also reported in the hep page).

Fig. 42 – Example of a web page automatically open after the click on the help button

After completion of the parameter configuration, the “Submit” button launches the experiment.

After launch of an experiment, it can result in one of the following states:

- **Enqueued**: the execution is put in the job queue;
- **Running**: the experiment has been launched and it is running;
- **Failed**: the experiment has been stopped or concluded with any error occurred;
- **Ended**: the experiment has been successfully concluded;

Fig. 43 – Some different state of two concurrent experiments

3.6.1 Re-use of already trained networks

In the previous section a general description of experiment use cases has been reported. A specific more detailed information is required by the “Run” use case. As known this is the use case selected when a network (for example the MLP model) has been already trained (i.e. after training use case already executed).

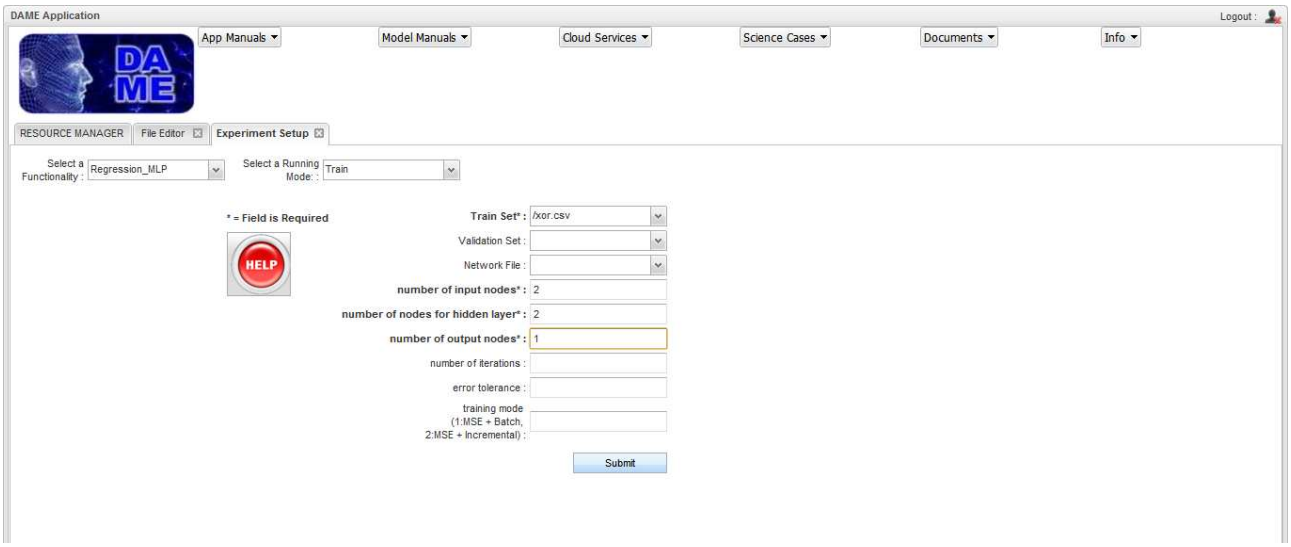


Fig. 44 – An example of Regression_MLP training case for the XOR problem

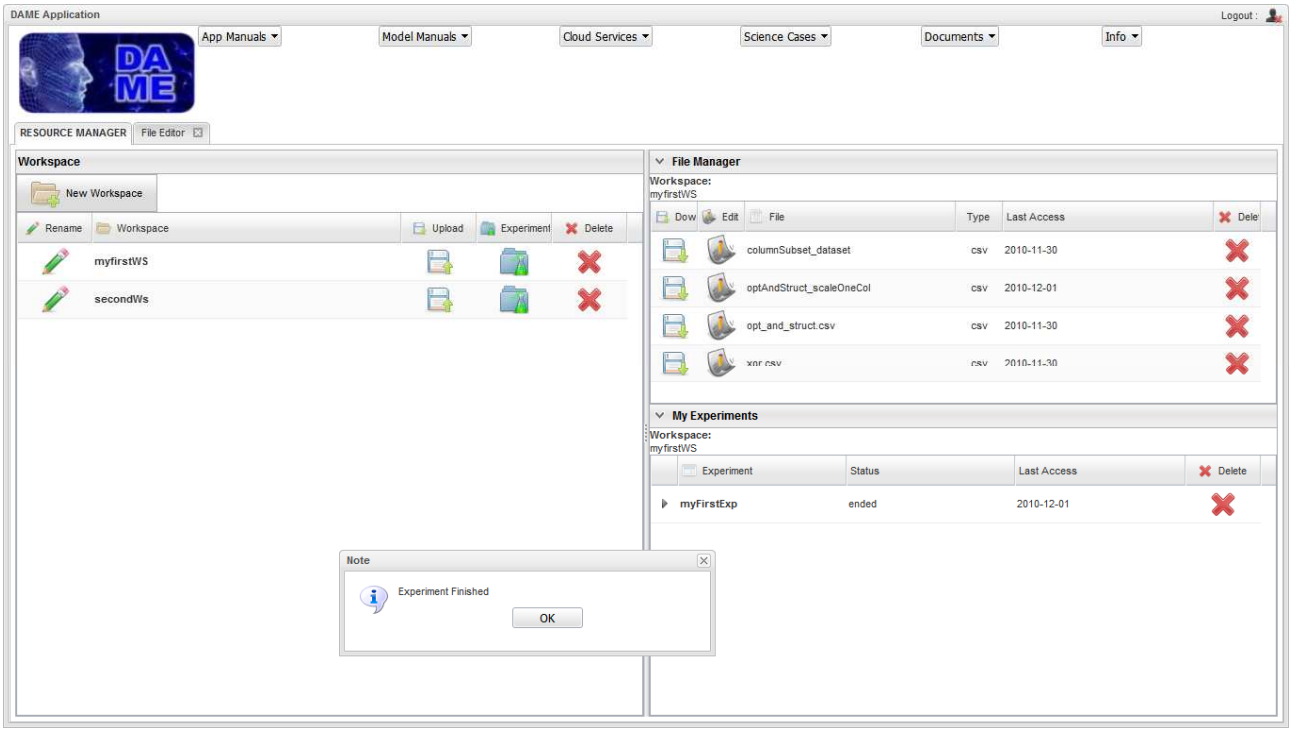


Fig. 45 – The status at the end of the XOR problem experiment

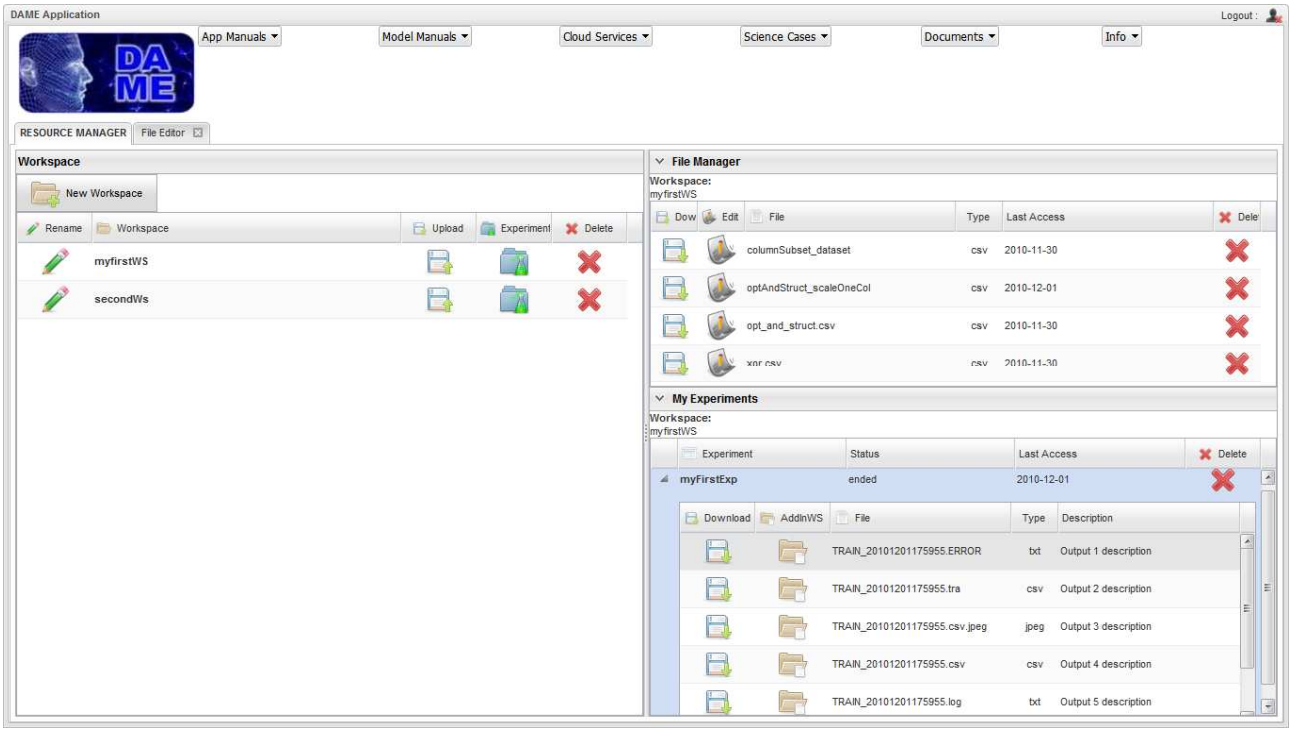


Fig. 46 – The list of output files after the XOR problem training experiment

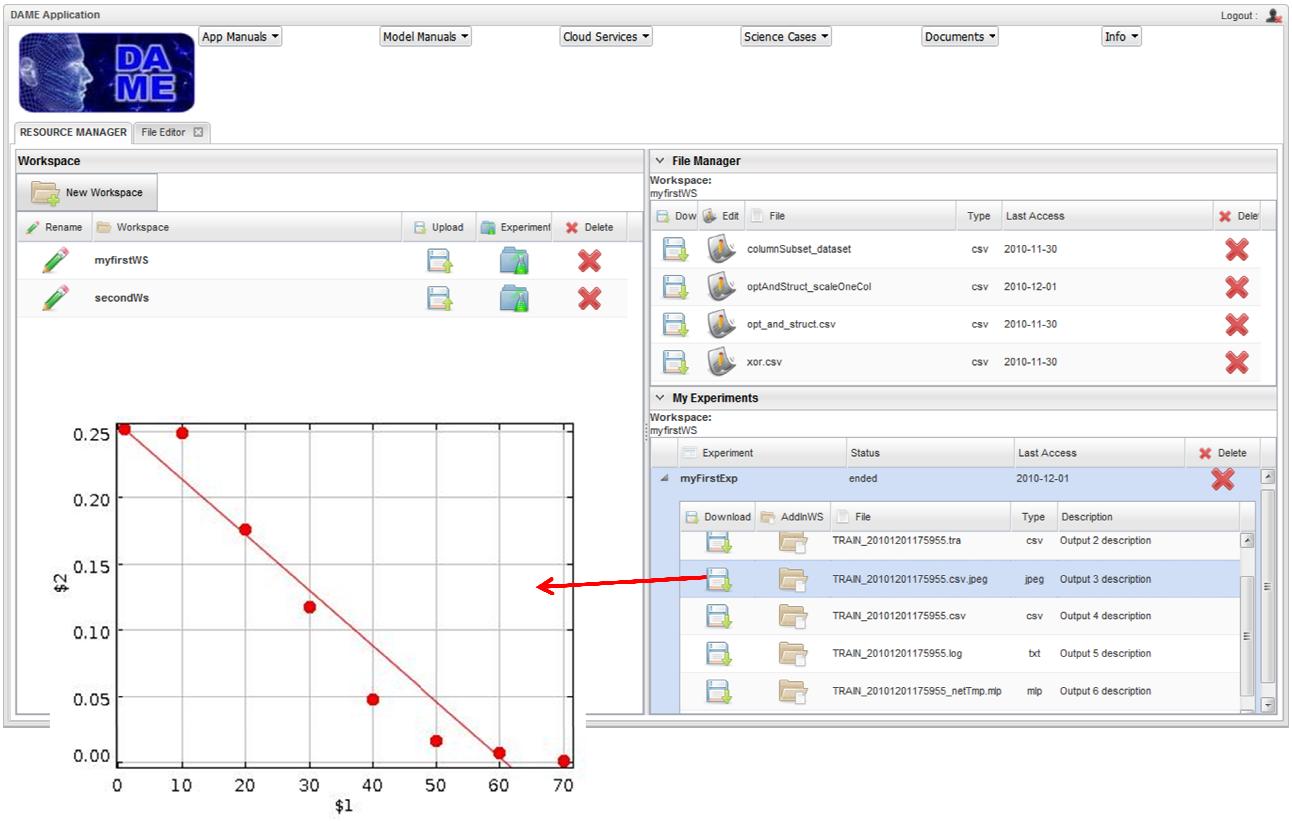


Fig. 47 – The training error scatter plot downloaded from the experiment output list (x-axis is the training cycle, y-axis is the training mean square error)

The Run case is hence executed to perform scientific experiments on new data. Remember also that the input file does not include "target" values. The execution of a Run use case, for its nature, requires special steps in the DAME Suite. These are described in the following.

As first step, we require to have already performed a train case for any experiment, obtaining a list of output files (train or full use cases already executed). In particular in the output list of the train/full experiment there is the file *outputFileName_netTrain.mlp*. This file contains the final trained network, in terms of final updated weights of neuron layers, exactly as resulted at the end of the training phase. Depending on the training correctness this file has in practice to be submitted to the network as initial weight file, in order to perform running sessions on input data (without target values).

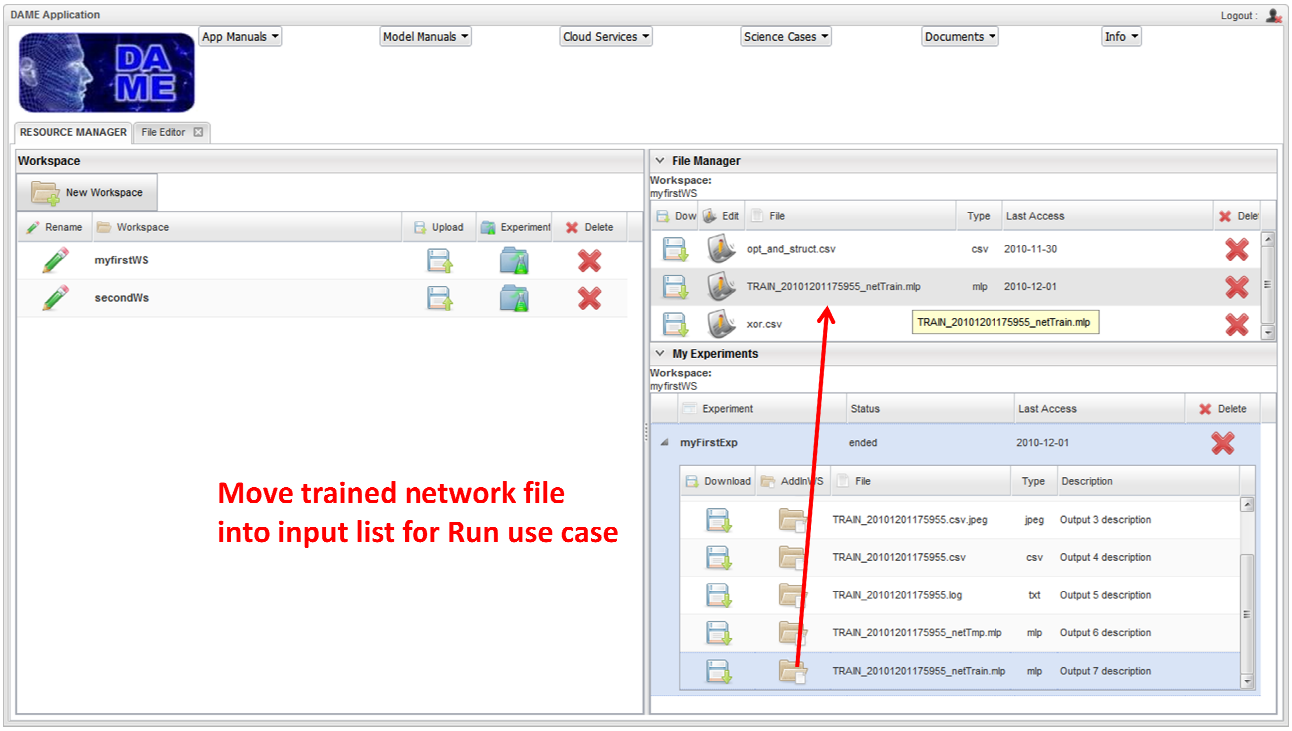


Fig. 48 – The operation to "move" the trained network file in the Workspace input file list

To do this, the output weight file must become an input file in the workspace file list, as already explained in section 3.5.4, otherwise it cannot be used as input of Run use case experiment, Fig. 48. Also, the workspace currently active, hosting the experiment we are going to do, must contain a proper input file for Run cases, i.e. without target columns inside.
So far, the second step is to populate the workspace file list with trained network and Run compliant input files and then to configure and execute the Run experiment (see Fig. 49)

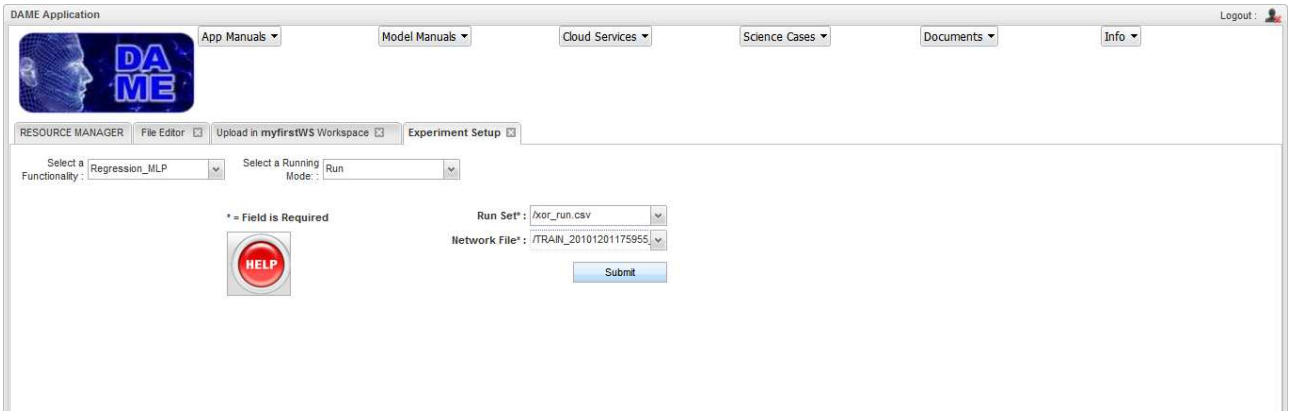


Fig. 49 – the configuration for the Run use case in the XOR problem

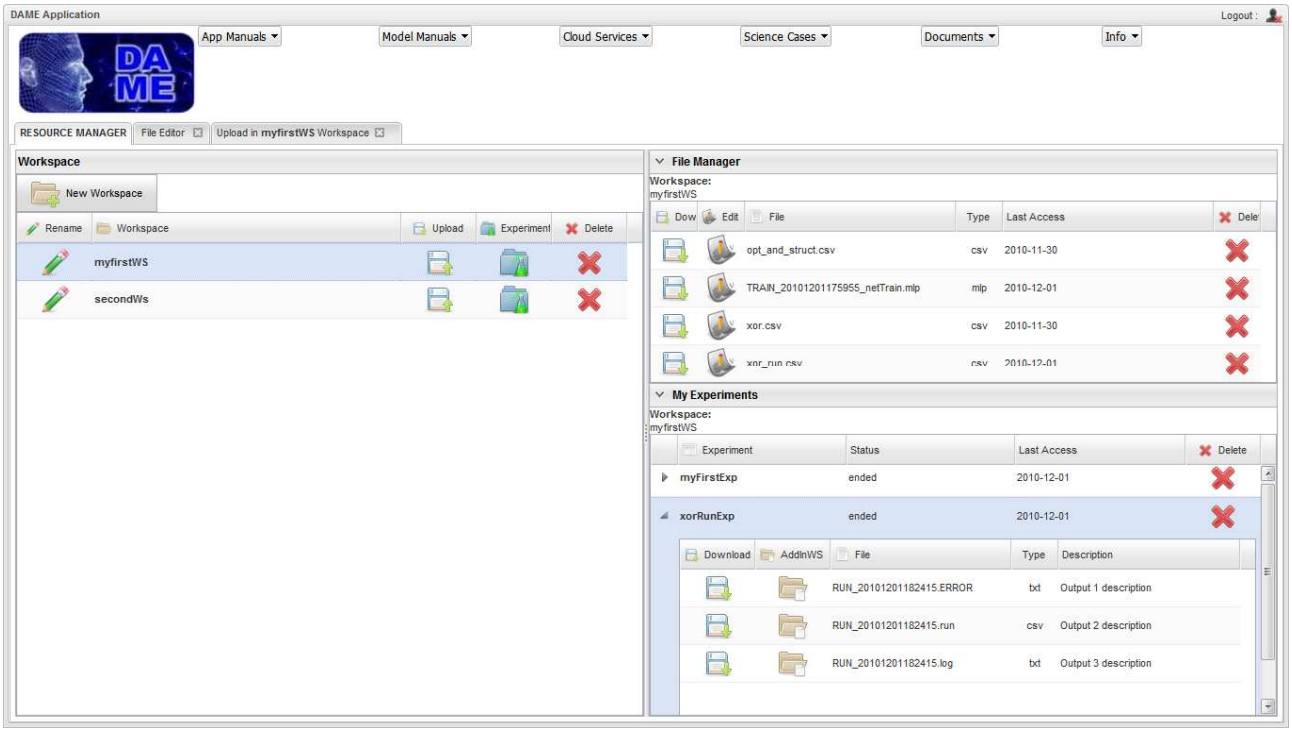


Fig. 50 – the output of the Run use case experiment in the XOR problem

At the end of Run experiment execution, the experiment output area should contain a list of output files, as shown in Fig. 49 and Fig. 50.

Also the same file *outputFileName_netTrain.mlp* should be selected as Network file input in case you want to execute another training (TRAIN/FULL cases) phase, for example when first training session ended in an unsuccessful or insufficient way. In this cases the user can execute more training experiments, starting learning from the previous one, by resuming the trained weight matrix as input network for future training sessions.. This operation is the so-called "*resume training*" phase of a neural network.

Of course, the same XOR problem could be also solved by using another functionality-model couple (such as Regression_MLPGA).
We remind the user to consult, when available, the related model specific documentation and manuals, available from the header menu of the webapp, the beta intro web page or the machine learning web page of the official DAME website.

Abbreviations & Acronyms

A & A	Meaning	A & A	Meaning
AI	Artificial Intelligence	KDD	Knowledge Discovery in Databases
ANN	Artificial Neural Network	IEEE	Institute of Electrical and Electronic Engineers
ARFF	Attribute Relation File Format	INAF	Istituto Nazionale di Astrofisica
ASCII	American Standard Code for Information Interchange	JPEG	Joint Photographic Experts Group
BoK	Base of Knowledge	LAR	Layered Application Architecture
BP	Back Propagation	MDS	Massive Data Sets
BLL	Business Logic Layer	MLP	Multi Layer Perceptron
CE	Cross Entropy	MSE	Mean Square Error
CSV	Comma Separated Values	NN	Neural Network
DAL	Data Access Layer	OAC	Osservatorio Astronomico di Capodimonte
DAME	DAta Mining & Exploration	PC	Personal Computer
DAPL	Data Access & Process Layer	PI	Principal Investigator
DL	Data Layer	REDB	Registry & Database
DM	Data Mining	RIA	Rich Internet Application
DMM	Data Mining Model	SDSS	Sloan Digital Sky Survey
DMS	Data Mining Suite	SL	Service Layer
FITS	Flexible Image Transport System	SW	Software
FL	Frontend Layer	UI	User Interface
FW	FrameWork	URI	Uniform Resource Indicator
GRID	Global Resource Information Database	VO	Virtual Observatory
GUI	Graphical User Interface	XML	eXtensible Markup Language
HW	Hardware		

Tab. 2 – Abbreviations and acronyms

Reference & Applicable Documents

ID	Title / Code	Author	Date
R1	"The Use of Multiple Measurements in Taxonomic Problems", in Annals of Eugenics, 7, p. 179--188	Ronald Fisher	1936
R2	*Neural Networks for Pattern Recognition*. Oxford University Press, GB	Bishop, C. M.	1995
R3	*Neural Computation*	Bishop, C. M., Svensen, M. & Williams, C. K. I.	1998
R4	Data Mining Introductory and Advanced Topics, Prentice-Hall	Dunham, M.	2002
R5	Mining the SDSS archive I. Photometric Redshifts in the Nearby Universe. *Astrophysical Journal*, Vol. 663, pp. 752-764	D'Abrusco, R. et al.	2007
R6	*The Fourth Paradigm*. Microsoft research, Redmond Washington, USA	Hey, T. et al.	2009
R7	Artificial Intelligence, A modern Approach. Second ed. (Prentice Hall)	Russell, S., Norvig, P.	2003
R8	Pattern Classification, A Wiley-Interscience Publication, New York: Wiley	Duda, R.O., Hart, P.E., Stork, D.G.	2001
R9	Neural Networks - A comprehensive Foundation, Second Edition, Prentice Hall	Haykin, S.,	1999
R10	*A practical application of simulated annealing to clustering.* Pattern Recognition 25(4): 401-412	Donald E. Brown D.E., Huntley, C. L.:	1991
R11	*Probabilistic connectionist approaches for the design of good communication codes.* Proc. of the IJCNN, Japan	Babu G. P., Murty M. N.	1993
R12	*Approximations by superpositions of sigmoidal functions.* Mathematics of Control, Signals, and Systems, 2:303–314, no. 4 pp. 303-314	Cybenko, G.	1989

Tab. 3 – Reference Documents

ID	Title / Code	Author	Date
A1	SuiteDesign_VONEURAL-PDD-NA-0001-Rel2.0	DAME Working Group	15/10/2008
A2	project_plan_VONEURAL-PLA-NA-0001-Rel2.0	Brescia	19/02/2008
A3	statement_of_work_VONEURAL-SOW-NA-0001-Rel1.0	Brescia	30/05/2007
A4	MLP_user_manual_VONEURAL-MAN-NA-0001-Rel1.0	DAME Working Group	12/10/2007
A5	pipeline_test_VONEURAL-PRO-NA-0001-Rel.1.0	D'Abrusco	17/07/2007
A6	scientific_example_VONEURAL-PRO-NA-0002-Rel.1.1	D'Abrusco/Cavuoti	06/10/2007
A7	frontend_VONEURAL-SDD-NA-0004-Rel1.4	Manna	18/03/2009
A8	FW_VONEURAL-SDD-NA-0005-Rel2.0	Fiore	14/04/2010
A9	REDB_VONEURAL-SDD-NA-0006-Rel1.5	Nocella	29/03/2010
A10	driver_VONEURAL-SDD-NA-0007-Rel0.6	d'Angelo	03/06/2009
A11	dm-model_VONEURAL-SDD-NA-0008-Rel2.0	Cavuoti/Di Guido	22/03/2010
A12	ConfusionMatrixLib_VONEURAL-SPE-NA-0001-Rel1.0	Cavuoti	07/07/2007
A13	softmax_entropy_VONEURAL-SPE-NA-0004-Rel1.0	Skordovski	02/10/2007
A14	VONeuralMLP2.0_VONEURAL-SPE-NA-0007-Rel1.0	Skordovski	20/02/2008
A15	dm_model_VONEURAL-SRS-NA-0005-Rel0.4	Cavuoti	05/01/2009
A16	FANN_MLP_VONEURAL-TRE-NA-0011-Rel1.0	Skordovski, Laurino	30/11/2008
A17	DMPlugins_DAME-TRE-NA-0016-Rel0.3	Di Guido, Brescia, Cavuoti	14/04/2010
A18	BetaRelease_ReferenceGuide_DAME-MAN-NA-0009-Rel1.0	Brescia	28/10/2010
A19	BetaRelease_Model_MLP_UserManual_DAME-MAN-NA-0011-Rel1.0	Cavuoti, Brescia	30/11/2010
A20	BetaRelease_Model_SVM_UserManual_DAME-MAN-NA-0013-Rel1.0	Cavuoti, Brescia	30/11/2010
A21	BetaRelease_Model_MLPGA_UserManual_DAME-MAN-NA-0012-Rel1.0	Brescia	30/11/2010

Tab. 4 – Applicable Documents

Acknowledgments

The DAME program has been funded by the Italian Ministry of Foreign Affairs, the European project *VOTECH* (Virtual Observatory Technological Infrastructures) and by the Italian *PON-S.Co.P.E.*

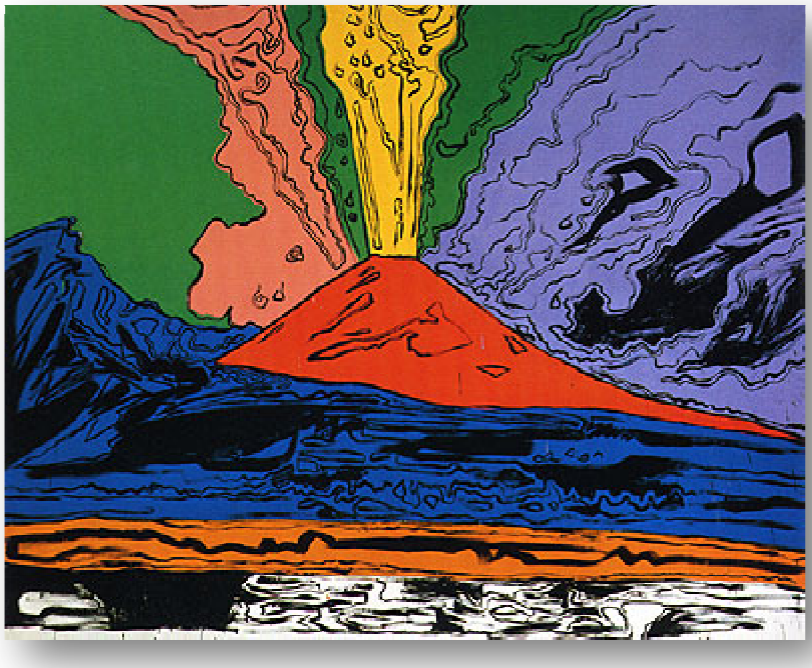
The current release of the data mining Suite is a miracle due mainly to the incredible effort of (in alphabetical order):

Stefano Cavuoti, Giovanni d'Angelo, Alessandro Di Guido, Michelangelo Fiore,
Mauro Garofalo, Omar Laurino, Francesco Manna, Alfonso Nocella, Bojan Skordovski

We want to really thank all actors who contribute and sustain our common efforts to make the whole DAME Program a reality, coming from University Federico II of Naples, INAF Astronomical Observatory of Capodimonte and Californian Institute of Technology.

Max

__oOo__



DAME Program
"we make science discovery happen"